

AD-A199 048

DTIC FILE COPY

①

SUMMARY OF PRODUCTS AND RECOMMENDATIONS



DTIC
ELECTE
AUG 11 1988
S H D

This is not an approved JLC document.

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

88 9 10 00

JOINT LOGISTICS COMMANDERS 4TH BIENNIAL SOFTWARE WORKSHOP

"ORLANDO II"

VOLUME I

SUMMARY OF PRODUCTS AND RECOMMENDATIONS

JUNE 1987

PRODUCED FOR THE
JOINT LOGISTICS COMMANDERS JOINT POLICY COORDINATING GROUP
FOR COMPUTER RESOURCE MANAGEMENT

THIS DOCUMENT WAS PRODUCED BY THE
POST DEPLOYMENT SOFTWARE SUPPORT SUBGROUP

Ms Shirley Peele, Chairperson	Navy (Surface)
Ms Lucille Cook	Navy (Air)
Mr Mel Weiner	Army
Capt Colin Gilyeat	Air Force (AFSC)
Mr Eldon Jensen	Air Force (AFLC)
Ms Myrna Olson	Marine Corps

This publication is required for official use only.
Other requests for this document must be referred to:

Commander
FCDSSA
Dam Neck
Virginia Beach, VA 23461

FOREWORD

Orlando II was the fourth in a series of biennial workshops focusing on relevant software support issues pertinent to Mission Critical Computer Resources (MCCR). The previous workshops, Monterey I & II and Orlando I, were instrumental in the identification of issues that could be addressed in Department of Defense (DOD) standards for the development of mission critical systems. One of those issues dealt with how to handle the problems associated with Post Deployment Software Support (PDSS). The central theme of Orlando II was "Solving the PDSS Challenge." Workshop selectees were assigned to one of eight panels. Each panel was assigned one particular PDSS problem area, and tasked with developing solutions. The panel's conclusions reinforced the fact that more cooperation is needed among the Services.

Specific panel topics were as follows:

- I. PDSS PLANNING DURING DEVELOPMENT
- II. FORECASTING PDSS RESOURCE REQUIREMENTS
- III. SOFTWARE CHANGE PROCESS
- IV. PDSS STANDARDS
- V. PDSS MANAGEMENT INDICATORS AND QUALITY METRICS
- VI. HUMAN RESOURCES IN PDSS
- VII. SOFTWARE TECHNOLOGY TRANSITION
- VIII. MCCR SECURITY

This volume presents a summary of the issues and recommendations of the eight workshop panels and is taken directly from the products provided by the panels without editorial comments or reinterpretations. Volume II of this report presents the workshop proceedings which provide the details of the panels' products, recommendations and guest speaker presentations.

Any questions concerning this material may be forwarded to:

Chairperson, PDSS Subgroup
Ms Shirley Peele
Fleet Combat Direction Systems Support Activity
Dam Neck, Code 82
Virginia Beach, VA 23461

Telephone: AUTOVON 433-7257 or (804) 433-7257

By	per letter	
Distribution/		
Availability Codes		
Dist	Avail and/or	Special
A-1		

TABLE OF CONTENTS

	<u>Page</u>
1. INTRODUCTION	1
1.1 Background	1
2. INDIVIDUAL PANEL EXECUTIVE SUMMARIES	3
2.1 Panel I - PDSS Planning During Development . . .	3
2.2 Panel II - Forecasting PDSS Resource Requirements	5
2.3 Panel III - Software Change Process	7
2.3.1 Panel IIIA - Modeling/Support Strategies	7
2.3.2 Panel IIIB - Configuration Management	9
2.4 Panel IV - PDSS Standards	11
2.5 Panel V - PDSS Management Indicators and Quality Metrics	13
2.6 Panel VI - Human Resources in PDSS	15
2.7 Panel VII - Software Technology Transition	19
2.8 Panel VIII - MCCR Security	21
3. PRODUCTS AND RECOMMENDATIONS SUMMARIES	23
3.1 Panel I - PDSS Planning During Development . . .	23
3.2 Panel II - Forecasting PDSS Resource Requirements	31
3.3 Panel III - Software Change Process	37
3.3.1 Panel IIIA - Modeling/Support Strategies	37
3.3.2 Panel IIIB - Configuration Management	41
3.4 Panel IV - PDSS Standards	47
3.5 Panel V - PDSS Management Indicators and Quality Metrics	51
3.6 Panel VI - Human Resources in PDSS	55
3.7 Panel VII - Software Technology Transition	63
3.8 Panel VIII - MCCR Security	67

LIST OF FIGURES

Figure 1	PDSS Process	73
Figure 2	PDSS Detailed Model	75
Figure 3	PDSS Planning Activities	77

1. INTRODUCTION

a. The Orlando II Workshop reviewed current Department of Defense (DOD) Post Deployment Software Support (PDSS) activities for Mission Critical Computer Resources (MCCR) and made specific recommendations to improve software support capabilities. Orlando II's purpose was to focus on those difficulties experienced by both Government and industry agencies in support of software intensive systems and recommend solutions for those problems.

b. Orlando II identified areas offering significant payoffs in terms of cost reduction, improved system reliability, and procedures which would streamline the PDSS process. In addition, the workshop reviewed the status of Orlando I Workshop recommendations, identified unresolved recommendations, and charted a course of action to complete unfinished beneficial recommendations.

1.1 Background.

a. As a result of the growth of digital computer resources in weapon systems, it was necessary to standardize the development process of those systems. In 1977, the Joint Logistics Commanders (JLC) instituted a Joint Policy Coordinating Group on Computer Resource Management (JPCG-CRM) to accomplish this task. The mission of the JPCG-CRM was to "coordinate and ensure consistency in the preparation of new and revised regulations and standards, to provide recommendations on critical resource areas, and to provide a focal point for coordinating standardization programs." To accomplish this mission, the JPCG-CRM organized a series of joint Government/industry workshops that would be attended by selected representatives who were experienced computer resource practitioners.

b. The first workshop, Monterey I, was held in 1979 at the Naval Post Graduate School at Monterey, California. Monterey I dealt primarily with software development and acquisition issues -- DOD policy, development standards, documentation standards, quality assurance standards, and acceptance criteria. Two years later at Monterey II these issues were reviewed. New areas of concern were explored -- computer resource configuration item selection, standardization and accreditation of computer architectures, software cost estimating, and software reusability. These workshops identified the importance of coordinated support for MCCR.

c. The third biennial workshop, Orlando I, was held in late 1983. Monterey I and Monterey II had focused on software development and acquisition. Orlando I focused on the support of MCCR after the initial development and deployment. The continuing and growing interest in the subject of post development and post

deployment software support led the JPCG-CRM to form the PDSS Subgroup in June 1986. The PDSS Subgroup mission states:

"...[the Subgroup] will identify, address, and resolve when possible, the problems and issues related to the maintenance and support phase of the life cycle."

One of the earliest requirements of the PDSS Subgroup was to:

"Prepare and conduct an Orlando II workshop to revalidate or further definitize existing problems and define new ones requiring resolution."

2. INDIVIDUAL PANEL EXECUTIVE SUMMARIES

2.1 Panel I - PDSS Planning During Development.

a. Panel I of the Orlando II PDSS Workshop was challenged to investigate and submit specific recommendations concerning the planning, policy, and budgeting of PDSS during development. It was determined that detailed consideration of these three areas are critical during the system development phase. For without adequate planning, supported by both policy and budget provisions, effective and timely PDSS of mission critical systems would be nearly impossible to achieve.

b. The "PDSS Planning During Development" panel oriented their deliberations to provide three products, each one addressing a single aspect, such as planning, policy, or budgeting of PDSS during development. Specific recommendations in each area were investigated and developed for submittal to the JLC via the JPCG-CRM PDSS Subgroup. Panel members determined that four, of the 15 recommendations carried over from the Orlando I Workshop, had been either previously implemented or were no longer pertinent. The panel then addressed the remaining 11 recommendations and arrived at a consensus concerning their resolution.

c. It was also established that significant planning, policy, and budget initiatives have taken place since Orlando I. Both the Air Force and the Navy have taken extensive life cycle management initiatives with the overhaul and revision of AFR 800-14 and the introduction of OPNAVINST 5200.28. The panel strongly recommends that all the Services follow similar suit. In particular, the Planning Subpanel strongly recommends that the JLC should sponsor a review of Naval Air Systems Command policy for applicability across all Services relative to facilities, lab asset management, and Force Activity Designator priority of system software support activity operations.

d. After much deliberation, the Policy Subpanel determined that significant improvement could be made to existing defense level acquisition supplements, data rights regulations, work breakdown structures, and project management guidance for nondevelopment item/commercial off-the-shelf initiatives. These changes were determined to be mostly near-term, low-cost actions with high return-on-investment.

e. The subpanel, challenged to identify specific budget recommendations, concluded that the Orlando I recommendation (to create a separate budget appropriation for PDSS) should be scrapped as infeasible and impractical. Recommendations were developed relative to PDSS cost identification and identification of software development costs during both development and modification.

f. Panel I identified as a general recommendation that all Services should implement an awareness program to the Air Force BOLD STROKE initiative. It was a unanimous conclusion that this would be an effective method to communicate and obtain necessary consideration of the PDSS challenge.

2.2 Panel II - Forecasting PDSS Resource Requirements.

a. Successful planning for forecasting resources in support of Mission Critical Computer Systems (MCCS) requires proper tools to support the decision making process. Techniques, with high levels of management confidence and support, must be developed to permit accurate resource forecasting and budgeting for software support activities. Cost estimation, as generally practiced in industry and Government, prior to the introduction of Software Cost Estimation (SCE) models or methods, was based upon ad hoc processes. Processes ranged from "best guess" to informed management and/or technical estimates, to a range of primitive to highly complex, semi-automated to automated computational methods. Unfortunately, adherence to these ad hoc practices continues at a significant number of software development and support facilities today.

b. Given today's defense environment of reduced budgets, dramatic growth in software requirements and corresponding software costs, increasingly complex systems, and DOD's reliance on software to support the "force multiplier through technology" concept, a more established, analytical, and acceptable (to management) approach to software cost estimation must be implemented. This requirement exists for both the acquisition and support of MCCS software.

c. After due deliberation, Panel II proposed that the Government immediately take positive initiatives to quickly institutionalize the use of a SCE methodology in the acquisition and support of MCCS software. The SCE methodology must be viewed as a management and technical tool which provides a readily understandable, quantifiable basis for establishing software cost, schedule, and resource (personnel and computer support) requirements.

d. Panel II's major recommendations were derived from the review, deliberation, coordination, and adjudication of five basic issues: PDSS forecasting problems, standard forecast model(s), model characteristics, model criteria, and requirements for further investigation and research (i.e., Research and Development).

e. The key recommendations of Panel II were:

(1) Establish, on a Service basis, a policy and implementation mechanism which directs a Constructive Costs Model (COCOMO)-like method to be used for forecasting software development and software maintenance resource requirements.

(2) Establish, on a Service basis, a standard software data collection initiative based on a supportive standard data definition initiative.

(3) Define and implement a management and technically based SCE methodology training program.

(4) Establish a Service oriented research program to insert new and evolving technology in the SCE method.

f. The long term goal, supported by these recommendations, was determined to be the achievement and adoption of a DOD-wide standard SCE method. Reference to the panel report provides supporting discussions and detail.

2.3 Panel III - Software Change Process.

2.3.1 Panel IIIA - PDSS Modeling/Support Strategies.

a. Panel IIIA (PDSS modeling/support strategies) had two major objectives: (1) modeling the software change process, and (2) identifying PDSS strategy alternatives. The panel reviewed the findings of Orlando I and when applicable, tried to take advantage of their earlier work.

b. Realizing that DOD has still not adopted a definition for PDSS, the initial task was to agree upon a definition. Although many alternatives were considered, the panel concluded that the definition of PDSS recommended by Orlando I remains correct and applicable. The panel recommends that the Orlando I definition be adopted and implemented by the DOD. The Orlando I definition of PDSS is as follows:

Post Deployment Software Support (PDSS) is the sum of all activities required to ensure that, during the production/deployment phase of a mission critical computer system, the implemented and fielded software/system continues to support:

- o its original operational mission,
- o subsequent mission modifications, and
- o product improvement efforts.

c. The Orlando I PDSS model served as a basis for initial discussions. The approach was to simplify the model to improve understanding and to make the model generic so that it would apply to all Services. The following conclusions were reached:

(1) The PDSS process model should reside within the total system support model.

(2) The PDSS process consisted of many activities which could be classified as either management, technical, or support functions.

(3) The PDSS process consisted of three phases: Phase I (Initial Analysis), Phase II (Software Development), and Phase III (Product Logistics). Figure 1 (page 73) is the high level view of the PDSS Process, while Figure 2 (page 75) depicts the PDSS Detailed Model. Phase II is the software development model contained in DOD-STD-2167. Phases I and III, which include primarily management and support activities, are new distinctions. The final model, which is simpler than the Orlando I model, clearly identifies the activities that occur in the PDSS process and provides a logical, and distinct, separation between each phase. The last consideration is important because Phase II is

frequently performed through contractor support while Phases I and III are most often accomplished by the Service. Additionally, the model incorporates logistics activities which are not incorporated in the Orlando I model or in DOD-STD-2167.

2.3.1.1 PDSS Contingency Model. The PDSS contingency model depicts how the PDSS process can be streamlined to satisfy extraordinary user requirements for rapid response (e.g., to correct faults that affect safety or a critical mission capability). The panel concluded that the PDSS contingency model was identical to the PDSS model. In other words, none of the activities in the PDSS model could be omitted to expedite the PDSS process. Instead, management could speed up the process by assigning appropriate priorities, eliminating unnecessary management controls, eliminating unnecessary tasks that are normally associated with a specific activity, or allocating additional resources.

2.3.1.2 PDSS Strategy Alternatives. The panel concluded that the management activities of the PDSS process must always be retained by the Government. The panel then examined factors which could impact the Government's ability to make alternative strategy decisions. Key considerations in the support strategy decision were: the volatility of the software, ownership of the software development facility (environment), and ownership of the software integration facility (environment). It was also concluded that the approved software support strategy and supporting requirements, to include the ownership of the development and integration facilities, must be reflected in the Computer Resources Life Cycle Management Plan (CRLCMP).

2.3.2 Panel IIIB - Configuration Management.

a. The Configuration Management (CM) subpanel was tasked to identify software and firmware related deficiencies in DOD CM directives and standards as they relate to PDSS activities, and to develop a recommended approach for implementing required changes. Additionally, the subpanel was tasked to develop basic procurement documents for the development of an automated standard software Configuration Status Accounting (CSA) system.

b. The subpanel conducted a detailed review of 13 major directives, standards, and specifications dealing with DOD CM policies, practices, and procedures. Although the review indicated that software CM issues were addressed to some extent in the majority of the documents reviewed, it was concluded that those documents were deficient and inconsistent with current PDSS philosophy and practices. This was not surprising, since the majority of the reviewed documents were issued in the early 1970's, long before many of the current software development and support philosophies were established. The subpanel also found that the reviewed documents were generally inconsistent in their relational approach to DOD-STD-2167, which is considered to be the guiding standard for all defense system software development efforts. The detailed changes recommended by the subpanel will be provided to the DOD Configuration Management Committee (DCMC). The DCMC has agreed to use these recommendations to initiate its planned overhaul of the area of DOD CM standardization.

c. The subpanel investigated the implications and requirements for developing a standard software CSA system. Issues addressed included how software CSA data could efficiently and practically be transferred from a developing activity to a support activity, methods and strategies to accomplish this transfer, and the trade-offs of various CSA system architectural approaches. Specific products that were developed included a specification of essential common data elements needed in any CSA system, and guidelines for writing a statement of work for the development of an automated software CSA system. The subpanel recommended that the JLC develop a formal handbook for use by DOD activities engaged in the development, procurement, or modification of a CSA system, and reaffirmed the need for a common automated software CSA system.

(Intentionally Blank)

2.4 Panel IV - PDSS Standards.

a. DOD-STD-2167 and DOD-STD-2168 provide a process for software development and quality assurance. They were established to be used in the development and acquisition of MCCR. They now require review to see if they should be modified to address PDSS issues.

b. Panel IV was tasked to review both DOD-STD-2167 and DOD-STD-2168 and determine what type of changes should be made for PDSS. Also, DOD-STD-1467 (an Army software support standard) was reviewed to determine if any Army specific requirements should be incorporated into DOD-STD-2167.

c. The panel discussed the PDSS environment and the status of the software development standards. They then recommended that several changes be made to the standards. The basic discussion focused on what should be changed and how the changes should be made. Three subpanels were formed to review these topics. The following is a summary of proposed change actions:

- (1) Describe the post deployment phase.
- (2) Define the preliminary software development activities.
- (3) Address modification to non DOD-STD-2167 developed items within a DOD-STD-2167 environment.
- (4) Change title to: "Defense Systems Software Development and Support."
- (5) Incorporate identified requirements from DOD-STD-1467 into DOD-STD-2167.
- (6) Incorporate identified requirements from DOD-STD-1467 Data Item Descriptions (DIDs) into DOD-STD-2167 DIDs.
- (7) Incorporate changes identified by subpanel reviews into DOD-STD-2167.
- (8) Incorporate changes to emphasize the software building process.
- (9) Add transition information to the Computer Resources Integrated Support Document (CRISD) DID.
- (10) Provide a means for the delivery of documentation for commercially available software in DOD-STD-2167.

d. The panel determined that one or more of the following options could be used to incorporate these proposed change actions into the software development standards.

(1) Modify DOD-STD-2167 in the following ways.

(a) Add an appendix to give top level guidance, provide the same information as the body of the standard while adding a PDSS perspective, or add an appendix to explain how to modify paragraphs in the body for PDSS.

(b) Rewrite paragraphs in the body of the standard by modifying existing paragraphs for PDSS or by adding shadow paragraphs.

(2) Develop a parallel PDSS standard.

(3) Develop a PDSS handbook.

The panel preferred option 1 of modifying the existing DOD-STD-2167. Two panel members felt a separate PDSS standard was needed.

e. The JLC should review the panels' recommendations for inclusion into the software development standards.

2.5 Panel V - PDSS Management Indicators and Quality Metrics.

a. The question is no longer whether management indicators and quality metrics are required, but how to institutionalize their use in the acquisition, deployment, operation, and maintenance phases of a weapon system and its support tools. Management indicators and quality metrics are essential if DOD and its industry partners are to turn the current DOD perceived state of software "witchcraft" into a science as defined by Lord Kelvin, when he said:

"When you can measure what you are speaking about, you know something about it. When you are unable to use a quantitative description, then your knowledge is meager and unsatisfactory."

b. Software engineering professionals generally recognize that the use of indicators and metrics provides visibility and control for management and product quality. Metrics are the potential cornerstone of a true systems engineering discipline. The proper use of indicators and metrics substantially increases the probability of developing and supporting a quality product within performance, cost, and schedule constraints. Without the use of quantitative metrics and indicators, product attributes such as mission effectiveness, reliability, availability, and maintainability are undefinable and therefore virtually unachievable in a cost effective manner. Without such indicators and metrics, possible weapon system warranties and effective software risk management techniques are not achievable, simply because there are no tools with which to understand the degrees of uncertainty of the system components.

c. A DOD management indicators and quality metrics program must cover the product, the processes, and all indicator/metric support tools. This program builds upon current Air Force (800 series) initiatives and embraces the sharing of other Service efforts. From this framework, the program covers improved ways to make metrics and indicators a by-product of the way we do business. For consistency, accuracy, completeness, and cost effectiveness this program must automate the metric gathering process.

d. DOD policy, directives, and standards need to incorporate metrics and management indicators to institutionalize the program. A multilevel phased training program must be established. Sharing of common indicator/metric tool sets and data banks across DOD agencies is required for the program to be cost effective. New research efforts must be established and funded to assure the metrics are kept current with ever changing computer and software technologies. A metrics information distribution center and clearing house is needed to promote industry and DOD cooperation. These efforts would also refine and develop better measurements as

newer technologies mature. In the PDSS phase metrics and tools must be transitioned from development to post deployment if redundancy and excessive maintenance costs are to be avoided.

e. It was the unanimous opinion of Panel V that a full-time, multiservice subgroup of the JLC JPCG-CRM be established to formalize the framework of the metrics program plan and oversee its implementation. Their discussions highlighted the fact that a lack of communication and coordination across the DOD and industry areas significantly retarded the sharing and use of our valuable engineering metric discipline. The Panel's final conclusion is summed up as follows:

"Across multiple DOD agencies, represented by this panel, better communication is required. Without this we have no leadership with which to forge a winning team. The user, academic, research and development, management, practitioner, contractor, and Government communities must be better integrated if we expect practical leadership to emerge. We must overcome this "data void" to further the software engineering discipline. Our national defense may be at stake. What objective is more vital?"

2.6 Panel VI - Human Resources in PDSS.

The objective of Panel VI was to define actions necessary to ensure the recruitment, retention, and training of knowledgeable software personnel to support PDSS. This panel was established as an outgrowth of the Orlando I Workshop.

2.6.1 Major Considerations/Discussion Points.

a. The panel recognized early in its discussions that this objective was very broad in scope and that the DOD personnel topic is a complex and multifaceted area that includes: people, organizations, and regulations. The panel reviewed the "Software Technology for Adaptable Reliable Systems (STARS) Functional Task Area Strategy for Human Resources" report, published by DOD in 1983, which identified six major subtask areas related to personnel and education.

b. Software related personnel subtask areas include:

- (1) Assessment of key populations,
- (2) Career structures, incentives, and mechanisms, and
- (3) Exchange programs.

c. Software related education/training subtask areas include:

- (1) Education programs,
- (2) Training programs, and
- (3) Learning aids.

Such strategy documents, however, are designed to provide only a conceptual planning approach. The Human Resources Panel was not in a position to tackle a detailed analysis of all these subtask areas, and decided to focus attention on more immediate problems and issues. It was also noted that individual agency initiatives, such as the Air Force BOLD STROKE Action Plan, were bringing management attention and understanding to the dominant role that software plays in weapon system effectiveness.

d. Project BOLD STROKE detailed four objectives for attacking software problems:

- (1) Awareness,
- (2) Education and training,
- (3) Personnel management, and
- (4) Future planning.

The thrust of such initiatives coincided with the discussions and recommendations developed by this panel.

e. It was estimated that, according to the Electronic Industries Association (EIA), the total demand for software by DOD will increase at a rate of 12 percent per year for the next two decades. The availability of personnel having requisite skills in computer and software engineering and/or system specific knowledge to support PDSS requirements will be a continuing problem. PDSS is, and will continue to be in the near future, a labor intensive activity. The availability of a qualified labor force is a significant determining factor in how a PDSS effort will be staffed. Direct hire authority, special salary rates, payment of relocation costs to first duty station, and accelerated training agreements have greatly enhanced Government's ability to attract entry level civilian engineers. Reduced hiring by private industry in 1986 and 1987 has also improved the applicant pool for potential entry level positions. However, new career management, educational, and training initiatives are needed.

2.6.2 Recommendations.

a. The Office of Personnel Management (OPM) is in the process of establishing a new software engineering job series for the civilian work force. This new series is the first step in establishing an expanded career ladder for computer scientists and software engineers. We recommend that the JLC support approval of proposed computer engineer (GS-8XX) and computer scientist (GS-1550) classification standards by OPM and request appropriate revision of the OPM X-118 engineering qualification standards.

b. Pay banding concepts, alternatives for simplifying the existing position classification, and pay systems have been implemented throughout DOD via various demonstration projects. This concept is incorporated into the DOD legislative proposal entitled Civil Service Simplification Act of 1986. We recommend that the JLC endorse proposed DOD legislation through appropriate channels, reiterating the need for greater management flexibility.

c. In addition to the high demand for software engineering skills, there are only a limited number of undergraduate computer engineering/software engineering degree programs available. In September 1983, the Educational Activities Board of the Institute of Electrical and Electronics Engineers (IEEE) Computer Society published a model curriculum program in computer science and engineering which defined curricula features and provided standards for developing new programs or modifying/upgrading established programs. We recommend the JLC support the model program and encourage the Software Engineering Institute (SEI) to market the concept to colleges and universities.

d. There currently exists in DOD a critical need for a consolidated and concise approach to software engineering training, a need to create awareness within DOD management of the mission critical PDSS software training requirements, and an assurance that appropriate training funds will be available. It is recommended that the JLC JPCG-CRM establish a subgroup, similar to the CSM Subgroup, to assess software training courses and Service needs. The tasking for this subgroup should include the development of an automated data base for tracking all current DOD software training courses.

e. Anticipated funding and manpower reductions resulting from Gramm-Rudman and other austerity measures have compounded the problem of maintaining sufficient personnel levels to meet PDSS support activities. We recommend that the JLC pursue a short-term solution to retain existing personnel levels by fencing off critical PDSS spaces and protecting them from potential cuts. This action would enable the Services to better maintain their mission readiness posture.

(Intentionally Blank)

2.7 Panel VII - Software Technology Transition.

2.7.1 Objective. The stated objective of panel VII was to identify policies/practices for transitioning necessary tools/methods and controlling their proliferation so that PDSS needs are met in a cost effective manner. In connection with this objective two panel tasks were identified:

a. Identify problems and recommend solutions for the insertion of support tools and new technologies into PDSS activities.

b. Identify problems and recommend solutions for the transition of operational software (tactical programs) from the developing to the supporting organizations.

2.7.2 Summary of Panel Findings.

a. Three perspectives/issue areas were used to address technology transition:

- (1) DOD Policies/Practices Issue Area.
- (2) Contractual Issue Area.
- (3) Software Tools and Environments Issue Area.

b. The panel prioritized the recommendations based on:

- (1) Ease of JLC ability to direct implementation.
- (2) Impact on the PDSS.
- (3) Ease of overall implementation.

The priority which resulted was:

- (1) Promulgate DOD Software Support Policy.
- (2) Establish PDSS Software Commonality Office.
- (3) Promulgate Software Support Environment Standards.
- (4) Improve Acquisition Regulation Support.
- (5) Promulgate DOD PDSS Policy.
- (6) Improve PDSS Training for Managers.
- (7) Modernize PDSS Tools/Technologies for Pre-Ada Systems.
- (8) Develop Ada Conversion Criteria.

(Intentionally Blank)

2.8 Panel VIII - MCCR Security.

a. The PDSS crisis is exacerbated by the lack of computer security in delivered systems. Retrofitting security into existing systems is costly and marginally effective. Specific issues are:

(1) Insufficient guidance for specifying and assessing MCCR security requirements.

(2) Lack of clear guidance for implementing and identifying MCCR security requirements.

(3) Inadequate capabilities for evaluating and certifying MCCR systems.

(4) Existing computer security R&D program does not adequately address MCCR requirements.

b. The key recommendations of Panel VIII are:

(1) Embed computer security requirements in DOD-STD-2167.

(2) Develop a computer security implementation guidebook.

(3) Establish organic Service certification and evaluation capability.

(4) Develop better guidance on identifying security requirements.

(5) Support an R&D program to:

(a) Adapt existing software engineering tools to enhance capabilities of computer security requirements in new systems and identify computer security weaknesses in existing systems.

(b) Develop automated tools and techniques to support trusted systems in the future.

(c) Develop efficient, effective MCCR security architecture.

(Intentionally Blank)

3. PRODUCTS AND RECOMMENDATIONS SUMMARIES

3.1 Panel I - PDSS Planning During Development.

a. Panel I of the Orlando II Workshop was challenged to investigate and submit recommendations regarding three specific areas. In order to accomplish this objective the panel divided into three subgroups. The three subgroups (identified as IA, IB, and IC) addressed the following:

(1) Identify, define, and prioritize PDSS activities that must be planned for during the software development phase. Develop a prioritized list of PDSS planning activities.

(2) Identify changes to current DOD regulations, standards, and directives to implement each aspect of planning identified above. Recommend specific modifications to DOD standards, directives, and regulations to implement each planning activity identified above.

(3) Identify methods of streamlining the budgeting process so that necessary software support resources are provided at the time of system deployment. Identify recommendations to improve the budgeting process.

b. Prior to breaking into subgroups, the panel began its deliberations by receiving several briefings that provided a framework for specific subgroups operations. Cognizant representatives from the three Services and industry presented comprehensive briefings and conducted active discussions relative to PDSS planning, policy, and budgeting activities that have taken place since the Orlando I Workshop. Subsequent to these briefings, the panel divided into subgroups that directed their attention to assigned tasks.

3.1.1 PDSS Planning.

3.1.1.1 Improvements to Acquisition Management of MCCR-Intensive Systems. The current system acquisition process does not adequately ensure proper life cycle computer resources supportability. The Project Manager (PM) mission/charter is limited to development responsibilities only and must be expanded to include the total system life cycle perspective. Deficiencies in MCCR acquisition occur as a consequence of insufficient MCCR expertise available to the PM from inception of system (e.g., poor Request for Proposal (RFP) preparation, no visibility for MCCR in milestone reviews).

3.1.1.1.1 Recommendation (Mid Term 2-4 years). Increase the visibility and accountability for MCCR issues by enhancing the major milestone review processes to include specific MCCR

questions and Defense Acquisition Board (DAB) members qualified to assess the responses. The RFP preparation process must be improved to preclude deficiencies in MCCR acquisition and long term supportability.

3.1.1.1.2 Recommendation (Mid Term). Expand the role and responsibilities of PM's Computer Resources Working Group (CRWG) by including trained personnel to provide comprehensive software engineering consultation in the following representative areas:

a. Use and extent of standard documents and DIDs commensurate with the complexity of system.

b. Feasibility of partitioning system functional requirements between hardware and software.

c. Long term MCCR supportability requirements (facilities, personnel specialties, support environment requirements).

d. MCCR cost estimates, including cost of any licensing or data rights considerations for Non-Developmental Item/Commercial Off-the-Shelf (NDI/COTS) resources and tools.

e. Capabilities of existing hardware and software suitability for meeting system performance requirements, in order to curtail proliferation of types of MCCR to be supported.

3.1.1.2 MCCR Cost Estimates. For a successful system, not only development cost, but cost and level of resources needed to support the system throughout its life cycle must be estimated during concept exploration and updated as system development progresses.

Recommendation. Identify the PM as the responsible individual for the assessment of total life cycle MCCR costs, and task the PM with the control of MCCR development costs.

3.1.1.3 PM Awareness of MCCR Requirements. Many implemented policies are not executed correctly because of the lack of training of the implementors.

3.1.1.3.1 Recommendations. When clarification is necessary, develop and issue handbooks and implementation guidance in parallel with the policy statement. Provide a point of contact to address user's questions, and whenever possible, augment information dissemination techniques through the use of teleconferencing, videotape, and newsletters.

3.1.1.3.2 Recommendation (Near Term). Develop a PDSS planning guidebook that ties required activities to major development milestones. Establish Figure 3 (page 77) as the JLC JPCG-CRM PDSS

During Development Activities. All Services should ensure that not-to-exceed milestone dates are identified and reflected, and that the following PDSS planning requirements are included in each of their respective life cycle management policies.

a. Designate and task the software activity prior to Milestone I.

b. Task the PDSS activity designated as principal in CRLCMP preparation, with coordination authorization after Milestone I.

c. Task the PDSS activity to perform or assist in performing independent verification and validation (IV&V) for MCCR software during system acquisition.

3.1.1.4 Modifications to DOD Standards, Directives, and Regulations Affecting PDSS Planning. DOD and Service level standards, directives, and regulations must be revised to enhance software visibility in system acquisitions, and streamline the acquisition process. Current DOD and Service policies do not adequately address the importance of software in systems and the large impact that software has on systems life cycle costs. Specifically, changes are required as delineated in the following subparagraphs.

3.1.1.4.1 Recommendation. The JLC should add the following PDSS policy to established Service policy (e.g., AFR 800-14, OPNAVINST 5200.28, etc.):

a. The PM shall identify the software support concept by Milestone II or before preparing the RFP for the development contract.

b. The selection of the support concept shall be based on total life cycle costs.

c. The development contract shall reflect support requirements (e.g., design constraints to enhance modification, licensing provisions, support software, etc.).

d. Program managers shall address estimated software life cycle costs and PDSS costs at major Service reviews.

3.1.1.4.2 Recommendation. The Services should sponsor a review of NAVAIRSYSCOM policy (NAVAIRINST 5230.9) for applicability concerning establishing software facilities at support locations early in the system life cycle, managing system support laboratory assets as part of the operational system, and assigning system software support activity Force Activity Designator (FAD) priority equal to the system being supported.

3.1.1.5 PDSS Policy. DOD and Service level policies must be revised to:

- (1) Enhance software visibility in system acquisitions, and
- (2) Streamline the acquisition process.

Current DOD and Service policies do not adequately address the importance of software in systems and the large impact that software has on system life cycle costs. Specifically, changes are required as delineated in the following subparagraphs.

3.1.1.5.1 Rights in Software. The need to perform software support for mission critical defense systems after deployment is not adequately addressed in the current rights in data policies of the Defense Federal Acquisition Regulation Supplement (DFARS) 52.227-7013.

Recommendation (Near Term). Recommend the Defense Acquisition Review (DAR) Council adopt a rights-in-software clause that reflects the intellectual property needs of software life cycle support.

3.1.1.5.2 DFARS Acquisition Documents. The DFARS requires a myriad of acquisition documents; e.g., Acquisition Plans and Justification and Approvals.

Recommendation (Near Term). Recommend the DAR Council modify the DFARS to properly reflect the reality of today's software intensive systems by requiring that software development and support issues be separately addressed in formal acquisition documents; e.g., Acquisition Plans and related documents as appropriate.

3.1.1.5.3 Management of Support Resources as an Integral Part of Systems Acquisition. Current DOD guidance and regulations are ambiguous with respect to acquisition and management of computer resources for support of mission critical defense systems. Specifically, Services are unclear whether to acquire the computer resources required to perform PDSS (generally commercially available computer resources) under the Information Systems directives (7920 Series) or Defense Systems directives (5000 Series).

Recommendation (Near Term). If it is necessary for DOD to have two sets of acquisition policies, one for defense systems (command, control, communications, intelligence weapons, tactical, and strategic) and one for automated information systems (business, data processing, and nontactical), then change the computer resources required to perform PDSS as parts of the systems they support for the entire life cycle of the system.

Also, review and modify acquisition policies to incorporate the development and production process discipline for post-Milestone III software activities.

3.1.1.5.4 MIL-STD-881A Revision to Address Software. The work break-down structure guidance specified in MIL-STD-881A does not emphasize nor recognize the magnitude of systems software costs. Application of MIL-STD-881A can result in no visible software costs and result in the inability of an acquiring activity to track software costs and schedule status.

Recommendation (Near Term). Modify MIL-STD-881A to reflect the terminology and methodology of DOD-STD-2167. Require software and associated activities/products to be identified to provide visibility, cost, and schedule status reporting and monitoring.

3.1.1.5.5 Policy to Require Computer Resource Joint Service Participation on Joint Programs. Regulations on joint programs do not require joint Service participation in planning PDSS nor do they provide guidance on funding and cost sharing for PDSS. Early joint planning could reduce software support costs if concepts such as centralized software support were analyzed.

Recommendation (Near Term). Require that Services incorporate a statement similar to the Navy policy in OPNAVINST 5200.28, Paragraph 19, which states:

"Joint Systems. For allied and joint Service systems in which the Navy is the lead Service, an interservice working group will be established. This group will ensure that analysis is performed to determine the optimum support approach for the life cycle; cost implications of major software support options; and the impact on operational needs, system life cycle costs, configuration management, interoperability, compatibility, and system integration. This group will document this analysis and make recommendations to the developing agency concerning the support approach."

3.1.1.5.6 Tailoring of DOD-STD-2167. Service policy and guidance on the use of DOD-STD-2167 does not emphasize tailoring this regulation to meet specific program characteristics. Service guidance is not available to allow acquiring activities to contractually require the minimum set of documentation necessary to organizationally support mission critical defense systems software.

Recommendation (Near Term). Services should emphasize the need to tailor the requirements of DOD-STD-2167 to allow for the cost effective acquisition of systems while balancing the cost of acquisition with effective software development and support requirements.

3.1.1.5.7 Improvements to Life Cycle Software Support Planning and Management. Typically, the acquisition process is set in motion before proper consideration of the impact of system and software design, support concept, and NDI/COTS on life cycle PDSS requirements/cost. There is little, if any, direct effort by PMs to determine a cost effective PDSS plan.

Recommendation. Develop Guidelines to provide information for PMs relative to life cycle PDSS support consequences resultant from utilization of NDI/COTS software rather than that conventionally developed under DOD-STD-2167.

3.1.2 PDSS Budgeting. As noted in the Orlando I report, funding of embedded software acquisition and support across the Services is provided through a variety of methods, using a mix of operations and maintenance, research and development, procurement, and modification appropriations. The Orlando I report advocated streamlining this funding process and establishing a separate "funding line" for PDSS. The panel found that the DOD Planning, Programming, and Budgeting System (PPBS) is largely driven by Congress, Office of the Management and Budget (OMB), Office of the Secretary of Defense (OSD), and the individual Service organizational structures.

Recommendation. While PPBS streamlining is desperately needed, pursuing it for embedded software alone would be infeasible and would fragment the funding of total systems.

3.1.3 Identification and Collection of Software Costs. Two recommendations of Orlando I dealt with the identification of software costs and appear to apply to the total system life cycle, including system development, system modification programs, and PDSS. In dealing with software costing, the subpanel divided the issue into separate categories:

a. System development and modification including both hardware and software, and

b. PDSS required to perform changes to tactical applications software programs that are not the result of companion hardware changes.

In the area of system development and modification the subpanel found a pervasive, overly simplistic view that by simply collecting software and hardware costs together would provide sufficient visibility into the development process. Further, the panel concluded that while certain benefits can be derived by collecting software cost information, it is not always practical to attempt to collect cost for all software configuration items in a modern weapon system. In the area of PDSS, it was concluded that in a major percentage of cases, costs are sufficiently

projected and tracked by the Services. The Services have taken major steps toward accomplishing the cost identification recommendation of the Orlando I Workshop.

Recommendation. Orlando II, Panel I-C, recommended that all Services develop and refine policies and instructions pertaining to software support similar to AF Regulation 800-14 and OPNAVINST 5200.28.

3.1.4 Productivity Improvement Resulting from Software Data Collection. Collection of software cost data will enhance pre and post deployment cost estimating and projections; identification of the reasons for cost growth; identification of future personnel needs; identification of areas to target for productivity improvement; and assessment of the impact of using new tools and standardization techniques.

Recommendation. The JLC should encourage the Services to continue to establish policy and procedures to:

- a. Collect PDSS costs for all weapon systems.
- b. Collect software costs, to a practical extent, for all software associated with systems development and modification that includes both hardware and software.

Regulations on joint programs do not require joint Service participation in planning PDSS nor do they provide guidance on funding and cost sharing for PDSS. Early joint planning could reduce software support costs if concepts such as centralized software support were analyzed.

3.1.5 General Recommendation. The PDSS Planning During Development Panel arrived at a unanimous conclusion that the best way to obtain necessary consideration for PDSS concerns was to make cognizant management aware of the problem. Therefore, Panel I strongly recommends that all Services develop and implement a program similar to Project BOLD STROKE of the USAF Systems Command. Project BOLD STROKE was viewed as a significant and timely activity that just may do more to solve the PDSS challenge than anything else.

(Intentionally Blank)

3.2 Panel II - Forecasting PDSS Resource Requirements.

3.2.1 Discussion.

a. Successful planning for the transition of new or modified systems into operational use requires proper tools to forecast resource requirements. Techniques which provide high levels of management, confidence, and support must be developed to permit accurate forecasting and budgeting for PDSS activities.

b. Panel II identified the following basic problems in the forecasting of PDSS resource requirements.

(1) Currently the estimation of PDSS resource requirements is largely unstructured and non-standard when viewed across the Services.

(2) There is not a designated Service level authority responsible for establishing guidelines for PDSS resource forecasting methodology/ies.

(3) Current forecasting techniques are not based on a valid historical data base for each PDSS center.

(4) There is no common definition of software development and PDSS terms or activities across DOD organizations.

(5) There is a lack of objectivity in current estimating techniques.

(6) Current techniques are often used to "back-in" to a pre-established, or approved budget, rather than to establish the actual required budget.

(7) Those using and/or inputting data for a forecasting technique are not adequately trained.

(8) The lack of a historical data base makes it difficult to predict change rates and resulting PDSS resource requirements during the development and support processes.

(9) The lack of a current, validated historical data base causes forecasting techniques to have limited acceptance by management.

(10) There are limited means for high level management to assess the impact of changes in funding levels, personnel allocations, or Government/contractor support ratios on the acquisition and support of software. The recommendations that follow were made by Panel II to provide the JLC JPCG-CRM with a course to follow, which will lead to a more effective method of forecasting PDSS resource requirements.

3.2.2 Recommendations.

3.2.2.1 Recommendation 1 - Established Service Method (Near Term). The JLC JPCG-CRM should support the establishment, on a Service basis, of a policy and implementing mechanism which directs a Constructive Cost Model (COCOMO)-like method to be used for forecasting software development and software support resources.

a. From panel discussions, it was found that all of the Services were predominantly applying some extensions of COCOMO. To date, the Army Life Cycle Software Engineering Community has adopted a COCOMO-based model called the Software Engineering Cost Model (SECOMO) as its standard for software resource forecasting. The Marine Corps is in the process of gaining acceptance for their COCOMO-based model as a standard for their forecasting of required software maintenance resources. The Air Force and Navy have not adopted a standard SCE model, but have used COCOMO techniques for some of their software forecast SCEs.

b. COCOMO's use as a de facto Service SCE model is in part attributable to its nonproprietary status. Its use is not restricted due to software data rights concerns. This, in turn, permits tailoring and common usage of the method by industry and Government with minimal restrictions and cost.

c. The immediate establishment of a policy and implementing mechanism, which directs that each Service utilize a COCOMO-like method, will help to quickly formulate a standard technique for forecasting PDSS resources.

d. The pertinent characteristics desired in a standard SCE forecasting model are as follows:

(1) The model must address activities and resources in a PDSS environment.

(2) The standard PDSS forecasting model should conform to DOD-STD-2167 and other related DOD standards.

(3) The model should support detailed cost, manpower, and schedule forecasting over the full life cycle.

(4) The model should be accurate, easily understood and accepted by management.

(5) The model should be adaptable to unique Service requirements.

(6) The model should have operational usage characteristics which are easy to use, portable, interactive, and contain easy to read output.

(7) The model should be well defined and supported by documentation, training, and Service implementation policy.

(8) The model should be flexible and extendable to allow incorporation of changes based on continuing research.

(9) The model's operational cost should be reasonable so that frequent reuse is not prohibitive.

3.2.2.2 Recommendation 2 - Standard Data Base (Near Term). The JLC JPCG-CRM should sponsor an initiative to establish, on a triservice basis, a standard software data collection initiative and a supportive standard data definition initiative.

a. Although the basic methodology structuring COCOMO is sound, obtainable results today will at best be a "ballpark" estimate, since modeled computational variables are based on multiple application industry data collected in the 1970's. Through application of specific software data collection, models can be statistically calibrated to more accurately predict costs, schedule, and other resource requirements. This, in turn, promotes more confidence in obtainable results.

b. Presently, there are no common data definitions of software development and PDSS terms and activities across DOD Services. By standardizing on a SCE technique, standard data definitions will be more easily formulated. Standard data definitions development is needed to establish data collection criteria. Also, a prescribed Work Breakdown Structure (WBS) for software data elements compatible with MIL-STD-881A Revision A (1 Dec 86) and DOD-STD-2167 must be defined to promote consistency for all data collection among systems. Data definition and collection initiatives on a triservice basis can produce the broadest maximum consistency for collecting software data from developing contractors, support contractors and in-house Government support.

3.2.2.3 Recommendation 3 - SCE Training (Near Term). The JLC JPCG-CRM should encourage the Services to define and implement a management and technically based training program to support the effective use, analysis, understanding, and acceptance of SCE method(s).

As with any new technology, SCE model training for technical personnel, nontechnical support personnel, and management is required. Without adequate training, nontechnical model users have difficulty understanding and implementing the model,

technical personnel have trouble inputting appropriate data, and management does not know the basis or the accuracy of results provided to them. Technically based training should help to minimize the "garbage-in and garbage-out" syndrome that results in a loss of confidence and credibility in modeled results; even where model algorithms may be accurate. Management based training promotes understanding and confidence necessary for management acceptance.

3.2.2.4 Recommendation 4 - Future Requirements (Near to Mid Term). The JLC JPCG-CRM, through the SEI and STARS Joint Program Office (JPO), should provide leadership toward the establishment of a Service oriented research program to develop and promote the insertion of new and evolving technology in SCE methodologies.

a. Off-the-shelf models such as COCOMO, while well defined in limited areas, do not address all software resource forecasting needs for each Service. Further research and investigation is needed in areas that expand existing SCE model capabilities, integrate the software model in the life cycle process, and determine resource forecasting needs that support merging software technologies. For long term research, the DOD should establish a central authority to support the upgrading of SCE methodology to reflect emerging software technology.

b. The pertinent areas desired for research and investigation to expand model capabilities are as follows:

(1) Tailor the SCE model capabilities to cover the software support organization's environment.

(2) Ensure that the model supports sensitivity analysis, "what if" analysis, estimation of confidence ranges, and identification of high risk approaches.

(3) Expand model coverage to estimate additional life cycle resource requirements such as: prototyping and requirements definitions; PDSS preparation; PDSS administration; acquisition management; facility management; contract management; system integration, test and evaluation; conversion; installation; training; data base administration; and computer resource requirements.

(4) Expand model coverage to complex software situations such as: incremental development; multiple versions; large, loosely coupled software complexes (combinations of operational, on-line support, and off-line support software); and mixtures of Government-supported and commercially-supported software.

(5) Develop better methods for estimating the amount of software to be developed or modified.

(6) Incorporate Ada language design methodologies.

(7) Add artificial intelligence and knowledge based systems characterization into the development and maintenance process of the SCE model.

(8) Add development and maintenance of embedded control systems with software using integrated circuit technology (e.g., Very High Speed Integrated Circuits) to the SCE model. Include appropriate characterization of the new types of hardware employed to develop and operate software, such as parallel processors and distributed networks, and incorporate new technologies such as reusable code repositories.

3.2.2.5 Recommendation 5 - Standard DOD Methodology (Long Term).
The JLC JPCG-CRM's long term goal should be to support the adoption of a standard DOD SCE model.

a. Without the focus created by a long term goal of adopting a DOD standard model, each Service is likely to establish diverging COCOMO-like methods for their use. The convergence of COCOMO-like methods for SCE models can stem from each Service sharing their modeling requirements, methods and tools, to help improve approaches for estimating their software resources. A DOD standard SCE model also helps to channel creative efforts into more productive areas by filling model voids. Costs are saved by minimizing duplication of effort, while limiting the use of a second model for only independent perspective auditor review. The adoption of a DOD standard SCE model promotes consistency amongst the Services for decision making, training, documentation, data collection, and comparison of costs.

b. The pertinent characteristics desired for a standard forecasting model still apply. Although there is much commonality amongst the Services, the creation of one standard DOD SCE model will require flexibility to cover options that may be specific to a Service. As new standard methodologies evolve from research and investigations, their placement in one single model is only advised if it does not make the model too cumbersome and complex. If it does, another standard model should evolve to complement the requirements not covered in the one model and to form one standard set of DOD models that handles all software requirements without any duplications.

(Intentionally Blank)

3.3 Panel III - Software Change Process.

3.3.1 Panel III A - PDSS Modeling/Support Strategies.

3.3.1.1 Objective 1. Identify the functions involved in the software support process and model that process.

3.2.1.1.1 Summary. After reviewing the Orlando I PDSS Model, the panel concluded that it was too complex to be adopted as a general process model at the DOD level; that it failed to address the relationship between initial software development and PDSS; and that it failed to emphasize the unique set of activities that distinguish PDSS from initial software development. Accordingly, the panel's objectives were to develop a simpler PDSS model, referred to as the Orlando II Model, by representing the process at a higher level of abstraction; to address the relationship between initial development and PDSS; and to emphasize the differences between PDSS and initial development. The panel first identified three functional categories that encompassed all PDSS activities: management, technical, and support. This approach differed from the Orlando I approach, which attempted to identify all PDSS functions: management control, configuration management, documentation, engineering, test, and software quality assessment. In addition, the software support process was modeled in accordance with the panel's previously stated objectives. Each activity depicted in the model was represented as belonging to one of the three functional categories.

3.3.1.1.2 Assumptions. The PDSS Process Model resides within the total system support model.

3.3.1.1.3 Conclusions.

a. The PDSS process consists of many activities which can be classified as either management, technical or support functions.

b. Figure 1 (page 73) depicts an overview of the Orlando II PDSS Process Model. The PDSS process consists of three phases: Phase I (Initial Analysis), Phase II (Software Development), and Phase III (Product Logistics). Phase II is the Software Development Model contained in DOD-STD-2167. Phases I and III, which include primarily management and support activities, are new distinctions. The final model is simpler than the Orlando I Model, clearly identifies the activities that occur in the PDSS process, and provides a logical and distinct separation between each phase. The last consideration is important because Phase II is frequently contracted while Phases I and III are most often performed by the Service. Additionally, the model incorporates logistics activities which are not incorporated in the Orlando I Model or in DOD-STD-2167.

c. The PDSS Contingency Model is identical to the PDSS model. In other words, none of the activities in the PDSS model can be omitted to expedite the PDSS process. Instead, management can speed up the process by assigning appropriate priorities, eliminating unnecessary management controls, eliminating unnecessary tasks, or allocating additional resources.

3.3.1.1.4 Recommendations.

a. That the definition of PDSS developed at Orlando I be approved as a DOD standard and implemented in appropriate regulations.

b. That DOD adopt a standard software support process model, based on the approach presented herein and similar to the PDSS Detailed Model shown in Figure 2 (page 75).

c. That control of the PDSS process be vested in the Government and all planning documents (such as CRLCMP, the Test and Evaluation Master Plan (TEMP), the CM Plan, the Quality Assurance Plan, and the Software Development Plan) specifically state the management control actions to be taken by the Government during support.

3.3.1.1.5 Anticipated Benefits.

a. A standard definition will promote common understanding of the PDSS process and the activities involved. For example, all three categories of software change will be considered as part of the software support responsibilities. This may have significant funding consequences and should be standardized within DOD.

b. It is difficult to standardize a process without first describing it or modeling it in some manner. The completed PDSS model will allow DOD to establish process standards. The approach used in the model clearly demonstrates the relationship between the software development process and the software support process.

3.3.1.1.6 Impact on PDSS if Not Implemented. Without a clear understanding of the PDSS process and the activities included therein, the PDSS process will be difficult to manage or standardize at the DOD level. The Orlando I Model is inadequate for that purpose because it is complicated and incomplete.

3.3.1.2 Objective 2. Identify software support strategy alternatives.

3.3.1.2.1 Summary. The panel recognized that there are two separate but closely related dimensions to this objective:

- (1) the level of support provided, and
- (2) the source of support activities.

The level of support dimension ranges from no support for software that will not change during the remainder of its life cycle to full support for volatile mission critical software. The source of support activities dimension addresses the issue of who should perform each software support activity (DOD versus commercial). The panel examined constraints which determine the set of strategy alternatives available to the Government. The key determinant for the level of support decision and for the source of support activities decision is ownership of the software support environment. This conclusion is based on legal and cost considerations. The software support environment consists of the software development environment and the system integration environment. Therefore, the focus of the panel's effort was on how these determinants affect the Government's support strategy alternatives.

3.3.1.2.2 Conclusions.

a. The PDSS strategy consideration must include a level of support decision and a source of support activities decision(s).

b. The key determinant for the PDSS strategy decision is ownership of the support environment.

c. The PDSS strategy decision is a fundamental decision which must be reflected in the CRLCMP. This decision affects subsequent decisions to obtain ownership of the software support environment which include the software development and integration environments.

3.3.1.2.3 Recommendations.

a. Those software support activities which are inherently management responsibilities should be retained by the Government. Examples of such activities include PDSS strategy planning; configuration identification/control/auditing; determination of quality attributes/standards and ensuring quality standards are achieved in the software product; and system level testing.

b. The PDSS strategy decision should be made early during initial development and should be reflected in the CRLCMP.

c. Resources necessary to accomplish PDSS should be procured based upon the PDSS strategy decision.

3.3.1.2.4 Anticipated Benefits.

a. As a matter of DOD policy, the PDSS strategy will be a conscious decision by Government made early during initial development. Software support resources will be obtained or not obtained based upon an approved strategy which is reflected in the CRLCMP.

b. Services will not become totally dependent upon the developing contractor because they failed to plan for a software support environment which is consistent with an approved PDSS strategy.

c. By retaining those PDSS activities which are inherently management responsibilities, Services can ensure that a software product remains supportable and quality standards are maintained throughout the life cycle.

3.3.1.2.5 Impact on PDSS if not Implemented.

a. Without a DOD policy regarding acceptable PDSS strategy alternatives, Services can become totally dependent on a contractor for continued software support of MCCR.

b. Failure to analyze the volatility of the software and the ownership of the support environment will result in limiting the PDSS strategy alternatives available to the Government.

3.3.2 Panel III B - Configuration Management.

3.3.2.1 Configuration Management Documentation Review.

3.3.2.1.1 Summary Discussion. The Configuration Management (CM) Subpanel was tasked to identify software and firmware related deficiencies in DOD CM directives and standards as they relate to PDSS activities, and to develop an approach for implementing required changes. The subpanel conducted a detailed review of 13 major directives, standards, and specifications dealing with DOD CM policies, practices, and procedures. The documents reviewed are listed below, together with a high level overview of recommended changes.

a. DODD 5010.19 (Draft) and DODI 5010.XX (when issued). These standards should be reformatted to reflect the software life cycle phases and should describe associated activities, reviews, and products within these phases for both hardware and software. This technique, used successfully in DOD-STD-2167, has proven very beneficial for both implementation and training. These documents must specifically address the operational phase for both hardware and software. For hardware, describe the maintenance; for software, describe the support in terms of policy direction and instruction for carrying out the associated activities, reviews, and products related to CM and the other supporting functions.

b. DOD-STD-2167 and DOD-STD-2167A. The life cycle software support planning requirements should be explicitly stated. The life cycle software support planning requirements should be documented in the Software Development Plan (SDP) and should include the orderly transition between life cycle phases. Appendix B of the SDP should provide details of CM activities during each life cycle phase, reflect requirements of DODD 5010.19, and provide for the transition of CM between life cycle phases.

c. MIL-STD-483A. A number of areas must be updated in order to be compatible and consistent with other associated military standards, properly address PDSS activities, and incorporate lessons learned from recent work. The most important of these items is the identification and incorporation of PDSS requirements early in the system life cycle and the early coordination with the designated PDSS activity.

d. MIL-STD-1521B. The primary recommendation for this standard is that the requirement for developing the Computer Resource Integrated Support Document (CRISD) should be emphasized by requiring a review of this document at the Physical Configuration Audit (PCA). The changes recommended for this document are consistent with DOD-STD-2167 requirements.

e. MIL-STD-490. There are several deficiencies in this document, the most notable of which is a lack of consistency with other associated standards. It is also recommended that the section on changes and revisions (3.3) be moved to DOD-STD-480A.

f. MIL-STD-499A. This standard needs a major revision to adequately address PDSS issues and several critical configuration management requirements.

g. MIL-STD-481, MIL-STD-482, and MIL-STD-1467. No changes required.

h. MIL-STD-2168 (MIL-Q-2168) and Joint Regulation. These items are not sufficiently advanced to warrant a review at this time. Also, the standard may be merged with DOD-STD-2167A.

3.3.2.1.2 Conclusions. Although the review indicated that software CM issues were addressed to some extent in the majority of the documents reviewed, they were deficient in terms of consistency with current PDSS activities and practices. This is not too surprising since the majority of these documents were issued in the early 1970s, long before many of the current software development and support philosophies were established. The subpanel also found that the reviewed documents were generally inconsistent in their relational approach to DOD-STD-2167, which is considered to be the guiding standard for all defense system software development efforts.

3.3.2.1.3 Recommendations.

a. The JLC request OSD, Director, Defense Data Management Office (DDMO), to initiate a major update of the DOD CM Plan, to include the formal, coordinated, and integrated review and update of all documents listed in the CM standardization area.

b. The JLC provide the detailed recommended changes in this report to the DCMC, with the recommendation that they be used to establish the initial formal update baselines for the applicable documents.

c. The JLC recommend to OSD that the PDSS Subgroup be tasked and funded by OSD to conduct the formal update of the CM documents, working under the cognizance of the Director, DDMO.

3.3.2.1.4 Anticipated Benefits.

a. Essential PDSS requirements would be properly integrated into the defense software development process, thereby providing significant life cycle cost benefits, as well as improved system accountability and maintainability.

b. DOD CM directives and associated standards would be current, and could be consistently applied to all defense software development efforts. This would significantly reduce overall defense software acquisition, procurement, development, test, and follow-on life cycle support costs.

3.3.2.1.5 Impact on PDSS if not Implemented. The existing inconsistent, incompatible, and outdated configuration management procedures, methods, and practices will continue to be applied to defense software development efforts, resulting in ineffective accountability and control of critical software baselines, reduced system reliability and maintainability, and paralleling increased development and life cycle costs.

3.3.2.2 Software Configuration Status Accounting.

3.3.2.2.1 Summary Discussion. The subpanel investigated the implications and requirements for developing a standard software Configuration Status Accounting (CSA) system. Issues addressed include the exchange of data among CSA systems, the transfer of software CSA data from a developing activity to a PDSS activity, methods and strategies to accomplish these exchanges/transfers, CSA report formats, and the tradeoffs of various CSA system architectural approaches. Specific products developed include a specification of essential common data elements, and guidelines for writing a statement of work for the development of an automated software CSA system.

3.3.2.2.2 Conclusions.

a. The subpanel reaffirmed the recommendation of the Orlando I CM panel that a common software CSA system be developed. This system would automate the software CM functions required by DODD 5010.19, DOD-STD-2167/2167A, and related standards. This system would be available for use by Government activities, and would be available as Government Furnished Equipment (GFE) to contractors working on Government software projects. The system should be developed from existing Service baselines to the extent practicable, and should consist of building blocks that may be replaced with commercial software tools already in place at PDSS activities. The system must be extensible and user tailorable to local site or project unique requirements, such as report formats, terminology, and security classification, and provide for the exchange of data among CSA sites. The system must support multiple site, multiple project, multiple host, and multiple participant CM activities from programmers to PMs. After development, the system could be turned over to one of the Service PDSS Software Commonality Offices proposed by the Orlando II Software Technology Transition Panel.

b. The subpanel also concluded that a military handbook was required to aid Government personnel engaged in developing their own software CSA systems. The handbook would address acquisition and procurement issues (RFP, statement of work, etc.), essential data elements, report generation, architectural design issues (distributed/centralized, etc.), host transportability, use of commercially available tools, data exchange, and other related issues that should be considered in the process of developing an automated software CSA system. As an initial step in this direction, the subpanel developed a contractual statement that would give a Government sponsor the necessary access to his contractor's CSA data, a list of essential software data elements, and a list of high level generic functional requirements for a common software CSA system.

3.3.2.2.3 Recommendations.

a. The JLC support the development of a common automated CSA system. This recommendation involves two complementary actions:

(1) The JLC fund the development of a formal system specification for the system.

(2) The JLC sponsor, promote, and oversee the development of the system. In this capacity, the JLC will solicit funding for the development effort from prospective user activities.

b. The JLC develop a military handbook, for use by DOD activities, covering all aspects of procuring, modifying, or developing an automated software CSA system.

3.3.2.2.4 Anticipated Benefits.

a. The development of a common automated software CSA system would provide the following benefits:

(1) Significantly reduced overall Government software development and maintenance costs. The availability of an adaptable set of integrated software CSA tools would save considerable R&D development funds. Also, the reduction in the number of multiple, functionally equivalent, CSA systems would significantly reduce software life cycle support costs.

(2) Provide a valid and consistent implementation of DOD software CM requirements across all the Services.

(3) The use of common CSA tools and procedures will significantly reduce overall training requirements.

(4) The use of common CSA tools, with data import/export features, will foster the exchange of data among user sites and provide a needed capability for remote site backup of critical information.

b. The development of a CSA handbook would provide the following benefits:

(1) For prospective users of the common CSA tools, the handbook would provide the information needed to adapt or otherwise tailor the available tools to their unique requirements.

(2) For those involved in developing their own CSA system, the handbook will shorten both the procurement and development times, and significantly reduce overall costs.

(3) Use of the handbook will promote the consistent implementation of DOD software configuration management guideline, procedures, and practices across all the Services.

3.3.2.2.5 Impact on PDSS if Not Implemented.

a. Without a common set of software CSA tools, the overall DOD funding needed for Service activities to independently develop and maintain their own systems will increase astronomically; unnecessary time will be consumed; training costs will soar; it will be increasingly difficult to transfer data among the various CSA systems; and there will be a greater potential for inconsistent implementations of DOD software CM practices.

b. Without the CSA handbook, CSA system development times and costs will increase, users will not have the benefit of lessons learned by others, interface problems will be more acute, and there will be a greater potential for the inconsistent application of DOD software CM practices.

(Intentionally Blank)

3.4 Panel IV - PDSS Standards.

a. Issues Raised and Pursued. One of the major tasks for Panel IV was to suggest PDSS related changes to DOD-STD-2167A and DOD-STD-2167. A discussion was held to make a list of perceived problems. No attempt was made to limit the ideas for consideration. The following list of changes was suggested:

- (1) Does not contain a strong pass down requirement.
- (2) Does not contain a strong traceability requirement.
- (3) Does not adequately address final preparation for delivery.
- (4) Does not adequately address the program build process.
- (5) Does not accommodate modification to existing documents.
- (6) There is no stress testing requirement.
- (7) Does not address degree of rigor required for software quality assurance during PDSS.
- (8) There is no definition of preliminary software development activities.
- (9) There is no definition of post software development activities.

b. Issues Raised but not Pursued.

- (1) Funding.
- (2) PDSS Contract procurement.
- (3) Firmware resolution.

c. Other Issues. Another panel task was to identify which of the requirements in DOD-STD-1467 should be incorporated into DOD-STD-2167.

3.4.1 Subpanel Reports. Panel IV was then divided into the following subcommittees.

a. Subcommittee IV A. Determine which requirements of DOD-STD-1467 should be incorporated in current software development standards.

Thomas Conrad
Jim Heil (Subcommittee Chairperson)
Kurt Krabbee
Dan Kvenvold

b. Subcommittee IV B. Identify changes to DOD-STD-2167 needed to incorporate PDSS considerations by analyzing DOD-STD-2167 to identify items that don't support PDSS.

Greg Bornako
Paul Byerley
Jim Parlier
Jane Radatz (Subcommittee Chairperson)
Jack Reichson
Wayne Sherer
James Steenwerth
Mae Stees

c. Subcommittee IV C. Identify changes to DOD-STD-2167 needed to incorporate PDSS considerations by analyzing the standard against the identified PDSS problems.

Karen Bausman
Rick Butler (Subcommittee Chairperson)
David Castellano
Ole Golubjatnikov
Charles Kelly
David Maibor
Lee Stewart

3.4.2 Recommended Changes to Standards. The three panels reviewed the suggested changes to the software development standards. The entire panel recommends the following changes be made to the standards:

- a. Describe the post deployment phase.
- b. Define the preliminary software development activities.
- c. Address modification to non-DOD-STD-2167 developed items within a DOD-STD-2167 environment.
- d. Change title to: "Defense Systems Software Development and Support."
- e. Incorporate identified recommendations from DOD-STD-1467 into DOD-STD-2167.
- f. Incorporate identified recommendations from DOD-STD-1467 DIDs into DOD-STD-2167 DIDs.
- g. Incorporate changes identified by subpanels reviews into DOD-STD-2167.
- h. Incorporate changes to emphasize the software build process.

i. Add transition information to the CRISD DID.

j. Provide a means for the delivery of documentation for commercially available software in DOD-STD-2167.

3.4.2.1 Options. It was determined that one or more of the following options could be used to incorporate these items into the software development standards.

a. Modify DOD-STD-2167 in the following ways.

(1) Add an appendix to give top level guidance, provide the same information as the body of the standard except from a PDSS perspective or add an appendix to explain how to modify paragraphs in the body for PDSS.

(2) Rewrite paragraphs in the body of the standard by modifying existing paragraphs for PDSS or by adding shadow paragraphs.

b. Develop a parallel PDSS standard.

c. Develop a PDSS handbook.

3.4.2.2 Conclusions. The panel preferred the first option, Option (a), that of modifying the existing DOD-STD-2167. Two panel members felt that a separate PDSS standard was needed.

3.4.3 Recommendation. The JLC should review the panel's recommended changes for inclusion into the software development standards, utilizing the methods set forth in Option (a).

(Intentionally Blank)

3.5 Panel V - PDSS Management Indicators and Quality Metrics.

3.5.1 Primary Recommendation of Panel V. The JLC PDSS should establish a Multiservice Multiphase Management Indicators and Quality Metrics Advocacy Program. Such a program should address the entire spectrum of policies, standards, guidelines, issues and activities that are necessary in applying such quality metrics and management indicators to DOD system developments. Without such high-level direction, the following individual recommendations will lack the urgency necessary to insure they are implemented across each component of the DOD. And without the adoption and implementation of the recommendations that follow, the DOD-viewed "black art" of software development we now pursue will not become a true engineering discipline in any of our lifetimes.

3.5.2 Recommendations.

3.5.2.1 Recommendation 1. JLC must establish and fund a JLC JPCG-CRM subgroup and program to foster the use of approved management indicators and quality metrics.

Description. Currently, management indicators and quality metrics are not being used consistently, even within individual Services, to improve cost, schedule and product quality. This is true even though there are numerous examples of specific programs within each of the Services which have gained significant benefits from their use.

3.5.2.2 Recommendation 2. Mandate the use of approved tailorable management indicators and quality metrics.

Description. Orlando II Panel V recommends the following existing guidelines to encompass the approved set to build upon today: AFSCP 800-43, dated 31 January 1986, and AFSCP 800-14 (DRAFT).

3.5.2.3 Recommendation 3. Revise appropriate DOD standards and DIDs to incorporate approved management indicators and quality metrics.

Description. Existing DOD standards and guidelines should be reviewed and revised to incorporate/mandate/require/tailor the use of JLC JPCG-CRM approved management indicators and quality metrics.

3.5.2.4 Recommendation 4. Require the use of proven, existing government owned automated indicator/metric tools.

Description. There are in existence today proven software tools that store, analyze, and partially automate the collection of the data that comprise management indicators and quality metrics. The use of these tools reduces the costs associated with

collecting and analyzing management indicators and quality metrics. Among the tools that are available are: The Automated Measurement System-AMS (RADC/COEE), Multistatic Analyzer Tool-MSAT (TECOM, Fort Huachuca), Complexity Analysis Tool-CAT (AMCCOM), Facility for Automated Software Production-FASP (NADC), Source Program Analyzer and Reporter-SPAR (NRL), and the Ada Measurement and Analysis Tool-AMAT (DRC).

3.5.2.5 Recommendation 5. Revise DOD-STD-2167 and DOD-STD-2168 (Draft) to incorporate error severity levels into the associated software problem report format.

Description. Revise the appropriate development standards and DIDs to incorporate error severity levels. Panel V recommends a scheme similar to that found in DOD-STD-1679A:

<u>Level</u>	<u>Description</u>
1	Fatal system error
2	One entire system function inoperative
3	One entire system subfunction inoperative
4	Minor code or documentation error
5	Miscellaneous (e.g., misspelling)

3.5.2.6 Recommendation 6. The JLC JPCG-CRM subgroup must seek feedback from users of approved management indicators and quality metrics to allow refinement and substantiation of same.

Description. To speed the process of improvement and enrichment of the original tailorable set of approved management indicators and quality metrics, the JLC JPCG-CRM subgroup needs feedback, both positive and negative, in order to improve the utility of these tools. Without this feedback loop, the management indicators and quality metrics may never achieve a status of usefulness and practicality, which is vital to their universal adoption.

3.5.2.7 Recommendation 7. DOD-STD-2167 software quality factor terminology definitions must be improved.

Description. There is a need to make the terminology for software quality factors more meaningful. This will allow upper management and less experienced technical support to better understand what is being measured during development and PDSS.

3.5.2.8 Recommendation 8. Establish a formal feedback loop from DOD maintenance activities back to the developing agencies.

Description. At a minimum, the feedback should describe:

- a. Which projects were successful (how, why, etc.).
- b. Which projects were failures (how, why, etc.).
- c. Lessons learned during maintenance and use.

3.5.2.9 Recommendation 9. Develop a multi-level guidebook detailing the recommended use of management indicators and quality metrics.

Description. The JLC JPCG-CRM subgroup should sponsor an effort to develop a management indicators and quality metrics handbook (guidebook) that integrates software management indicators (AFSC Pamphlet 800-43), software quality indicators (AFSC Pamphlet 800-14), software reliability measures (e.g., RADC-TR-87-XX, Guidebook for Software Reliability Prediction and Estimates), and software quality measures (e.g., RADC-TR-85-37, Specification of Software Quality Attributes).

3.5.2.10 Recommendation 10. Establish a comprehensive research and development (R&D) program for management indicators and quality metrics.

Description. Such an R&D program should encompass the following aspects, at a minimum:

- a. Survey current use of management indicators and quality metrics.
- b. Gather feedback on the use of management indicators and quality metrics.
- c. Procure automated tools and associated documentation.
- d. Study future areas for management indicators and quality metrics use.
- e. Support the use of automated management indicators and quality metrics in future programming support environments.
- f. Experiment with new/revised management indicators and quality metrics.
- g. Experiment with adding indicator/metric automated functions into DOD approved preliminary compilers and environments.

3.5.2.11 Recommendation 11. Emphasize the requirements definition phase throughout the software system life cycle.

Description. A much stronger emphasis should be placed on requirements definition and tracking in the system/software life cycle. The major benefits to be derived and potential problems to be avoided dictate this approach. Two implementation methods, both of which are already automated, can be employed: requirements traceability and rapid prototyping.

3.5.2.12 Recommendation 12. Mandate the use of tailorable management indicators and quality metrics during acquisition and PDSS procurements of DOD software.

Description. All agencies/organizations that have software based Computer Software Configuration Item (CSCI) development, testing and/or support requirements must be able to mandate tailorable software quality factors, criteria and metrics that are reflected in both development and PDSS contracts.

3.5.2.13 Recommendation 13. Establish a multilevel management indicators and quality metrics training program.

Description. Current management indicators and quality metrics technology exists, but is being practiced inconsistently within DOD. A key aspect of this is the awareness of this technology by acquisition managers and PDSS management.

3.5.2.14 Recommendation 14. Establish a multiservice management indicators and quality metrics central data/tool/reference bank.

Description. The JLC should mandate the collection of quality metric data, and deliver this data to the central management indicators and quality metrics data bank. This process of collection should use automated tools to the maximum extent possible. All tools shall also be deliverable. The metric data may be used by acquisition agencies to assure higher quality products. The PDSS agency will use the data to refine their resource requirements and to track software activity through life cycle completion.

3.6 Panel VI - Human Resources in PDSS.

3.6.1 Introduction. The panel recognized very early in its discussion that the human resources objective was too broad in scope and that the DOD personnel situation is a complex and multifaceted area which includes people, organizations and regulations. The panel reviewed the "Software Technology for Adaptable, Reliable Systems (STARS) Functional Task Area Strategy for Human Resources" report, published by DOD in 1983, which identified six major subtask areas related to personnel and education. The Human Resources Panel was not in a position to tackle a detailed analysis of all these subtask areas identified in the STARS report and decided to focus their attention on more immediate problems and concerns such as those outlined in the Air Force BOLD STROKE Action Plan. Project BOLD STROKE detailed four objectives to attack software problems:

- a. Awareness.
- b. Education and training.
- c. Personnel management.
- d. Future planning.

The thrust of such initiatives coincided with the discussions and recommendations developed by the Human Resources Panel.

3.6.2 External Constraints Issues. Personnel policies designed for software personnel are subject to numerous external pressures and constraints. The issue of human resources has been addressed from the standpoint of improving the ability of the Government to attract and retain knowledgeable software engineers and to maximize their proficiency through proper training and incentive programs. The authority and financial resources necessary to build an adequate staff has been assumed as given. It is recognized that the trend is toward heightened austerity and that the expectations may not be realized. Past efforts to justify resource requirements have been largely unsuccessful, mainly because the magnitude of the estimates has been disturbingly high, and the software support community has not been successful in gaining credibility for its estimation technique. Further, the impact of inadequate resources on operational readiness of the Mission Critical Defense Systems (MCDS) has not been convincingly portrayed to the decision makers for several reasons. First, it is impossible to forecast what kinds of failures will occur and to what extent they will degrade a systems' capabilities. Secondly, as each year goes by with the software support organizations funded to only a fraction of the stated requirements, there is no noticeable short range consequence. Additionally, the long range implications of continued inadequate staffing are too ephemeral to gain support in an environment of scarcity. Even the argument

that the workload is increasing because the number of MCDS is increasing has been unsuccessful. While the efforts to acquire additional resources must not abate, it is clear that an aggressive marketing and education program is necessary to achieve even as modest a goal as relief from anticipated manpower reductions.

3.6.2.1 Primary Issue. In a time of steadily decreasing manpower ceilings, it is unrealistic to seek higher authorizations. Therefore, it is increasingly important to seek to protect the existing authorizations from further erosion. Job series which require critical skills tend to be the series which experience the greatest turnover. The normal technique for implementing reductions in authorized slots is through attrition by eliminating vacant positions first. Were that to continue, the software support activities, because of the higher than average turnover, would be unfairly penalized. Such reductions would be burdensome with a stable workload; with an expanding workload it is intolerable.

3.6.2.2 Proposed Solution. Protect the critical software engineering skills from further cuts by fencing off the existing spaces.

3.6.2.3 Projected Benefits.

- a. Reduced reliance on costly contractual support.
- b. Retention of an in-house cadre for the preservation of corporate memory and maintaining technical expertise.
- c. Retention of trained, knowledgeable employees to avoid disruption of on-going system support.

3.6.2.4 Final Recommendation. The JLC actively promote the exemption of positions for software support of MCDS from manpower reduction actions.

3.6.3 Career Management Issues.

- a. The availability of a highly skilled computer and software engineering work force to support MCCR requirements is a vexing problem and will continue to impact how the PDSS effort will be staffed. Technical requirements should drive the PDSS staffing mix, and the mix should be made available through proper planning and implementation. PDSS is, and will continue in the near future to be, a labor intensive activity. The demand for software by the DOD is anticipated to increase at a rate of 12 percent per year for the next two decades, according to the EIA. Currently all Services are building PDSS staffs using primarily electronic engineers (GS-855), computer scientists (GS-1550), computer

specialists (GS-374), and to a small degree mathematicians (GS-1520). There are two major sources from which activities recruit for these skills:

(1) Recent college/university graduates for entry level positions GS-5/7.

(2) Experienced engineers and professionals.

b. The DOD has been faced with the constant challenge of recruiting sufficient numbers of engineers to meet its growing mission needs, especially in the MCCR area. DOD agencies have recognized that they needed to address this problem through the development of specialized and cost effective recruiting techniques, as well as designing innovative programs that would provide alternate sources of trained PDSS personnel. The use of direct hire authority, special salary rates, accelerated training plans, payment of interview and relocation costs to first duty station for critical skills occupations has greatly enhanced the DOD's ability to attract entry level civilian engineers.

c. Innovative recruitment initiatives such as the joint Air Force Logistics Command (AFLC) and University of Dayton Reentry Program is just one example of programs implemented within DOD activities to provide additional sources of engineering talent to support MCCR requirements.

d. 1986-1987 private industry hiring reductions have also improved the DOD's recruitment posture, though agencies are still experiencing difficulties in attracting experienced professionals especially for certain geographic locations. DOD work force trends indicate that attrition rates have dropped significantly since 1984. Turnover rates for computer professionals, particularly software engineers, still tend to be higher than other occupations. Computer professionals and software engineers are a multiple industry resource, and therefore have a variety of alternative opportunities. Their jobs are not extremely sensitive to supply and demand forces within particular industries because if one industry is in a slump, computer skills can often be transferred to another one that is prospering.

3.6.3.1 Personnel Retention Issue. The panel noted that the DOD is not positioned to be competitive in recruiting and retaining experienced PDSS personnel, therefore limiting our ability to meet the existing and projected software engineering requirements. The panel agreed that current personnel systems are cumbersome and that Government agencies need greater flexibility in assigning rates of basic pay in order to recruit, motivate and retain a well qualified work force. New career management procedures are also required.

3.6.3.1.1 Proposed Solution. Pay banding concepts, which are an alternative of simplification of existing position classification and pay systems, have been implemented within the DOD through various demonstration projects. This approach has been incorporated into a DOD legislative proposal entitled "Civil Service Simplification Act of 1986". Such legislation would allow the Naval Demonstration Project (i.e. pay banding concepts) to be incrementally expanded throughout the Federal work force in a controlled, measured and budget neutral manner. Other benefits are that it ties pay and retention to performance and is open to any occupation, activity or geographic area. Also included are changes to special salary rate provisions which would allow for special rates in a greater variety of circumstances, increase the available rate range when necessary and permit the hiring of individuals covered by special salary rates at a rate above the minimum established for that special rate range. The proposal would also permit the payment of recruitment or retention bonuses based on continued service agreements.

3.6.3.1.2 Final Recommendation. Although currently available avenues have enhanced the Government's ability to hire entry level scientists and engineers, we recommend that the JLC endorse the proposed DOD legislation through appropriate channels, reiterating the need for greater flexibility in rewarding the efforts of our senior level PDSS personnel.

3.6.3.2 Personnel Classification Issue. The panel discussions also surfaced the problem that the Federal Government is unable to assign civilian personnel having requisite skills in computer and software engineering to appropriately classified and structured positions. A triservice initiative chaired by the Navy developed a proposed classification standard covering computer engineering (8XX) and a revision of the computer scientist (1550) series. These new standards were submitted to OPM for review and approval. The DOD is expecting OPM to officially release this new computer engineering standard shortly. In conjunction with these new classification standards, OPM should take steps to revise the X-118 Qualification Standards to incorporate the computer engineering series. Current qualification standards do not address electronic and software engineering course work under their basic requirements.

3.6.3.2.1 Proposed Solution. Amend the X-118 qualification standards to include the computer engineering series. Modify basic requirements by inserting additional course areas relevant to electronics and software.

3.6.3.2.2 Final Recommendation. We recommend that JLC request appropriate revisions to the OPM X-118 qualification standards to incorporate computer or software engineering course work and reflect the new technologies.

3.6.4 Education and Training Issues. Improving the productivity of software engineers requires new ways of thinking and reasoning about software and better methods of producing it. To gain intellectual control over the software production process and become more productive and efficient, the DOD is aspiring to make the production of software less labor intensive and more technology intensive. The use of these new technologies requires users to be better educated and trained. Panel discussions began by defining education and training. Education is a long term activity based on fundamentals, and designed to build a foundation of knowledge and reasoning abilities. Training is a short term activity with a specific goal, and builds upon the educational foundation. The panel agreed that education fell into two categories:

- 1) Initial development of skills required to begin a software engineering career (i.e. Bachelor's Degree), and

- 2) Continuing education requirements to keep abreast of advancing technologies.

The challenge to educators is to provide the appropriate foundation for software engineers, so that the expected rapid advancements in technology can be used effectively after relatively short training periods.

3.6.4.1 Educational Curricula Issue. There are currently very few undergraduate curricula which provide the PDSS community with entry level professional software engineers. To increase the number of qualified software engineers, the number of software engineering educational programs must increase. To achieve this goal, we must work against the enormous inertia of an education system that does not respond quickly to new educational needs.

3.6.4.1.1 Proposed Solution. In September 1983, the Educational Activities Board of the IEEE Computer Society published a model curriculum program in computer science and engineering which addressed curricula and guidelines for the development of faculty, administration and material resources. The primary goals of the model program were:

- a. To provide and define curricula features of undergraduate programs in computer science and engineering.

- b. Provide standards of comparison that could be used to guide the development of new programs or the modification and upgrading of established programs.

- c. Provide standards for accreditation from the Accreditation Board for Engineering Technology (ABET).

d. Provide guidance to academic administrators concerning the level of commitment needed to support a program.

The DOD contract that established the SEI specifically mentions education as a mission of the Institute, saying:

"It shall also influence software engineering education curricula development throughout the education community."

Its proper role is to serve as a focal point and catalyst to influence software engineering curricula. Through its Education Division, the SEI should take the lead in marketing the IEEE model program in Computer Science and Engineering to respective colleges/universities in order to increase the number of new software engineers.

3.6.4.1.2 Projected Benefits. An undergraduate software engineering curricula would provide for a work force that is better equipped academically to support PDSS and be prepared for the anticipated transition to a technology intensive activity in the next decade.

3.6.4.1.3 Final Recommendation. JLC refine and market through SEI a model for a computer engineering/software engineering curriculum, such as the IEEE proposal, which would enable colleges and universities to readily implement and expand software engineering programs to adequately prepare students for such professions.

3.6.4.2 Training Issue.

a. There currently exists in the DOD community a critical need for a consolidated and concise approach to software engineering training and an increase of awareness at the middle management level of the need for such training. The training of our software engineers basically falls into two categories. The first of these is the continuing education/training of those individuals already working in the software area. The second and perhaps the more critical is the problem of cross-training individuals from other technical disciplines into that of software engineering.

b. There are several problems facing the DOD community relative to meeting these training needs. In many instances, middle level managers are not aware of the impending need to train their people to meet the PDSS challenge. In those organizations where the need is recognized, the manager is unaware of the numerous training courses already in existence. Often times, new training courses are developed "on the fly", duplicating those already in existence, finding out after the fact that the course has missed the mark relative to meeting their specific needs. The cross training problem is even more acute when the manager does

not have a structured mechanism for identifying and selecting those individuals or courses available to meet the larger task of career field changes. Without solutions to these problems, the projected DOD software personnel shortfall will remain unanswered and the PDSS challenge will not be met.

3.6.4.2.1 Proposed Solution. The solution to the above problems consists of many pieces.

a. Create a DOD-wide, mandatory training program to educate and raise the awareness level of DOD middle level managers. The Air Force project BOLD STROKE is a first step in this direction.

b. Create a data base of all current DOD and commercial software training courses. This data base should be implemented on electronic media and as a minimum should consist of an abstract describing the contents of the course.

c. Investigate a mechanism of updating and making available to the DOD, Government, and industrial community easy access to this data base.

d. Develop a mechanism to mandate cross-training of selected DOD personnel to the software engineering career field.

e. Establish guidelines and procedures for selecting those individuals for software cross-training programs.

f. Ensure training funds are available to meet the mission critical PDSS software training requirements.

3.6.4.2.2 Final Recommendations. Charter a joint Service and industry ad hoc group to assess PDSS training courses and Service needs, define a consolidated approach to software engineering training, create awareness in DOD management of software training and funding requirements, and develop an automated training data base.

3.6.5 Summary of Recommendations.

a. Establish a new software engineering job series (GS-8XX) for the civilian work force and request revision of OPM X-118 Qualification Standards for professional engineering series.

b. Adopt alternative position classification and pay systems (i.e. "pay banding") by supporting the DOD legislative proposal entitled Civil Service Simplification Action of 1986.

c. Refine and market a model for computer engineering and software engineering curriculum.

d. Task an ad hoc group to:

(1) Define a consolidated approach to software engineering training.

(2) Create an awareness in DOD management of software training and funding requirements.

(3) Assess available training and Service needs.

(4) Develop an automated data base.

e. Protect existing manning levels by "fencing off" critical PDSS spaces.

3.7 Panel VII - Software Technology Transition.

3.7.1 Discussion - Summary. The products and recommendations that resulted during the Panel VII deliberations were resolved in panel planning sessions to produce a consensus report. The three perspectives/issue areas that were used to address technology transition were:

- a. DOD Policies/Practices.
- b. Contractual.
- c. Software Tools and Environments.

Eight recommendations were developed in these three subject areas. They are summarized in the following sections.

3.7.1.1 DOD Policies/Practices.

3.7.1.1.1 Issue - DOD PDSS Policy. DOD-level policy is needed to explicitly address PDSS within the system development life cycle. The current overall system acquisition policy is less than adequate and seriously outdated. This policy (DODD 5000.29) and its implementing instructions provide guidance to both DOD and industry and should be maintained in an up-to-date status.

Recommendations. Specific actions which should be undertaken include:

- a. Develop and issue a DOD Instruction to implement the policy described in the current version of DOD Directive 5000.29.
- b. Strengthen the OSD oversight function (individual) for cognizance over software development and support decisions.
- c. Update and reissue DOD Directive 5000.29 to emphasize PDSS consideration during the acquisition process.

3.7.1.1.2 Issue - DOD Software Support Policy. There is currently no uniform DOD policy used in contracting for support software. Most contracts only address requirements for the acquisition of operational software. These contracts typically fail to specify software requirements related to life cycle support.

Recommendation. DOD-STD-1467, Software Support Environment, (18 January 1985), should be reviewed, modified (if applicable), approved, and promulgated on a DOD-wide basis. The review of DOD-STD-1467 should also consider including PDSS technology transition requirements that must be fulfilled to prepare the PDSS for system turnover acceptance and support. DOD (OSD) and the

Services should establish a comprehensive triservice review of DOD-STD-1467 and promulgate its extended use throughout DOD.

3.7.1.1.3 Issue - PDSS Training for Managers. DOD and contractor PMs developing systems with PDSS requirements do not thoroughly understand the software development process, the software life cycle, and the impact of supportability issues on the final products.

Recommendation. DOD and contractor PMs developing systems with PDSS requirements need an understanding of PDSS issues to adequately plan and execute pre-PDSS activities. Therefore, PDSS training will be necessary for PMs not thoroughly familiar with PDSS and related technologies. A three-level training program is proposed.

- Level 1 - Short videotape.
- Level 2 - One day tutorial.
- Level 3 - Two day tutorial.

3.7.1.1.4 Issue - Software Support Environment Standards (CAIS Implementation). There is a proliferation of software support environments and like tools within these environments throughout the DOD. Service direction and policy should be developed and effected as soon as possible in order to establish the DOD support base to encourage rapid implementation by contractors and agencies.

Recommendation. The DOD-STD-1838, Common APSE Interface Set (CAIS), is due for printing and distribution in February 1987. This version of CAIS will provide host and operating system transportability of tools used in PDSS activities. It will address the problem of a multiplicity of specific hardware and operating systems used at PDSS organizations. It will enhance technology transition by providing standard interfaces to plug tools into PDSS support environments thus allowing easy use of new tools on existing or new hardware/operating systems.

3.7.1.2 Contractual - Acquisition Regulation Support. The JLC needs to support the DOD Acquisition Regulations which include the DAR, Federal Acquisition Regulations (FAR), and DOD FAR Supplement. There are two concerns with the current DOD data rights policy:

- (1) Contractors are unwilling to utilize their most sophisticated tools or development efforts if they may have to deliver those tools, with unlimited rights, to the Government;
- (2) Entrepreneurial companies are unwilling to do business with DOD for fear of losing competitive advantage.

Recommendation. It is necessary to develop or to adjust the Acquisition Regulations so that state-of-the-art tools are available for PDSS. This may require selectively requesting unlimited rights on procurements; it may require that PDSS facilities have the option to mandate that deliverable software be supportable by existing tools. Once Acquisition Regulations that encourage the transition of technology into the PDSS environment have been developed, the policy must be communicated to PMS, to the PDSS community, and to DOD contractors.

3.7.1.3 Software Tools and Environments.

3.7.1.3.1 Issue - PDSS Software Commonality Office. The separation of the Services, the organizational and command separation within each Service, the alignment of PDSS organizations along acquisition program lines, and the concentration on immediate operational problems, all inhibit the identification, procurement, and widespread distribution of common PDSS tools, methods, and processes.

Recommendation. Each Service should establish at the command level, a PDSS software commonality office with the following charter:

- a. Identify, evaluate, procure, and distribute tools and methods to users at PDSS activities.
- b. Provide centralized support for user assistance, consolidation of user requirements, and resolution of software problems.
- c. Provide coordination between the Services and raise the level of visibility of PDSS concerns.

3.7.1.3.2 Issue - Modernization of Tools and Technology for PDSS of Pre-Ada Systems. PDSS requirements will significantly increase without a commensurate increase in resources causing the software crisis. DOD is attacking this problem for future systems with Ada technology; however, a majority of systems supported by PDSS for the next 15-20 years will not be in Ada. As a result, the Ada productivity improvements will not be realized by PDSS activities for some time, and potentially PDSS will not have the resources for adequate support during this transition period.

Recommendation. DOD (OSD), in addition to aggressively pursuing the Ada technology effort, should also pursue an other-than-Ada software engineering technology improvement program for PDSS technology improvement. The program should include increased tasking to the STARS program and the SEI.

3.7.1.3.3 Issue - Ada Conversion Criteria. It is generally recognized that systems implemented in Ada will be much easier and hence, less costly to maintain. However, because Ada is only now being required for developing systems, there is a large inventory of software in existence that is not in Ada, but which will have to be supported for the next 20 to 30 years. The reduction of this pre-Ada inventory is clearly desirable, however, it is not easy to determine when and if a given system should be considered for upgrade to Ada.

Recommendation. To allow conversion decisions to be made in a logical manner, it is necessary that criteria be established that will allow cost effectiveness to be established for the upgrade alternative.

3.7.2 Prioritization of Recommendations.

a. The panel prioritized the recommendations based upon:

(1) Ease of JLC ability in directing the implementation of the recommendations,

(2) Impact on the PDSS, and

(3) Ease of implementation.

b. Naturally, this priority algorithm was based on the experience of the panel participants, and may need to be altered by the JLCs based on current realities. The priority which resulted was:

(1) Promulgate the DOD Software Support Policy.

(2) Establish a PDSS Software Commonality Office.

(3) Promulgate Software Support Environment Standards.

(4) Improve Acquisition Regulation Support.

(5) Promulgate DOD PDSS Policy.

(6) Improve PDSS Training for Managers.

(7) Modernize the Tools and Technologies for PDSS of Pre-Ada Systems.

(8) Develop Ada Conversion Criteria.

3.8 Panel VIII - MCCR Security.

3.8.1 Findings/Discussion.

3.8.1.1 General. The single largest improvement in MCCR security can be achieved by integrating security requirements into the system engineering discipline. However, additional tools are required to satisfy and maintain the higher levels of trust necessary for more critical applications. The panel emphasized in the strongest possible terms that major improvements in system security can be made by including security requirements at the beginning of a project. This may allow the use of existing software tools to satisfy both security requirements in addition to "conventional" software development requirements. Satisfying security requirements is a requirement definition process and a rigorous application of sound systems engineering disciplines including a comprehensive quality assurance program.

3.8.1.2 Subtask A.

a. The identification of computer security requirements is dependent on the system application. For information processing systems, a secure system is one that guarantees the integrity of and proper access to the information. For process oriented systems, such as a weapons control system, security means ensuring that the weapon is trustworthy and will perform as intended; it is not inadvertently fired, it goes where it is supposed to, it is aimed correctly, and it is resistant to in-transit countermeasures. For control systems, such as a navigation system, the security aspect may be that the system always works (reliability).

b. Four issues were raised with respect to a PDSS Center:

- (1) Certification and accreditation of a system over its life cycle.
- (2) Accreditation for an existing (nonaccredited) system.
- (3) The impact of hardware maintenance for a secure system.
- (4) The impact of secure software maintenance and distribution.

c. Descriptions of certification and accreditation are as follows:

CERTIFICATION is the process of ensuring that an operational system precisely satisfies specified (security) criteria.

ACCREDITATION is a determination by proper authority, the Designated Approving Authority (DAA), that the operational system works well enough so that the operational need for the system outweighs the operational risk associated with system deployment when evaluated against the certification criteria.

The application of these definitions revealed three states for a system:

- (1) Deployed; not certified or accredited.
- (2) In development; security requirements not identified.
- (3) New starts; certified and accredited.

d. MCCR security activities for PDSS should be chosen in a manner that is independent of but takes into consideration applicable certification criteria. The set of activities to be applied to PDSS is determined by the state of the mission critical computer system. The variants of the set are the system's state of certification and accreditation, and the PDSS activities necessary to satisfy MCCR security requirements. The list of activities include:

- (1) Security certification and accreditation determination (all states).
- (2) Security enhancement assessment plan (deployed and in process states only).
- (3) Security accreditation plan (all states).
- (4) Security certification package (all states).
- (5) IV&V documentation.

3.8.1.3 Subtasks B and C.

a. Current regulations, guidelines and policy directives associated with security and mission critical systems deal primarily with information processing security and do not adequately address process security. The principle current regulation regarding computer security is DOD-STD-5200.28 ("Orange Book"). The Orange Book provides certification requirements and criteria for general purpose operating systems that must support DOD information security policy; it requires interpretation for application and it does not provide a complete baseline for certifying or accrediting process control systems that require a different interpretation of the term "security".

b. The Orange Book is a standard. Guidance for its use and application are inadequate. The Orange Book does provide a baseline for the derivation of criteria for other information processing security applications and for process control security applications. This baseline is provided by the theory and rationale contained in it, but other interpretations must be developed for data base, network, and process control application areas.

c. The support environment that exists in a PDSS center is usually a general purpose computer system whose resources are applied to the problem of providing life cycle support for a mission critical system. Such a system falls into the category of an information processing system for which the Orange Book criteria are applicable. However, further guidance is needed and the Naval Research Lab (NRL) Report entitled "An Approach to Determining Computer Security Requirements for Navy Systems", by Carl Landwehr and H. O. Lubbes, is an example of this guidance.

3.8.1.4 Subtask D.

a. Research and development efforts from which near term, mid term, and long term benefits could accrue were identified. In the near term (applicable to deployed and in development systems), the following efforts are suggested:

(1) The development of security specific testing tools and methods that include penetration packages, regression test support, stress testing, and code analysis tools.

(2) The definition and development of a standard evaluation process.

(3) The adaptation and application of existing software engineering tools.

b. Those projects for which mid term (three to five years) can be expected include:

(1) Security modeling for the solution space, the threat, and the necessary analysis.

(2) The application of knowledge based technology to the automation of the software development process supporting the transition between development life cycle phases while preserving the complete traceability and providing (semi) formal verification for the system.

c. Long term benefits can be expected from hardware and architecture efforts. The panel noted that mid and long term benefits would be realized for PDSS of new starts.

3.8.1.5 Subtask E. Metrics that could be applied to the determination of the extent to which security requirements are met included:

a. The extent to which a (disciplined) development approach was followed.

b. The extent to which the specific security evaluation criteria are satisfied.

c. Based on the application of code analysis tools and techniques, code quality, the presence or amount of "dead" code, and the complexity of the code.

d. The extent to which the system is modularized and the degree to which security critical code is isolated.

e. The anticipated amount of difficulty to accredit the system as a function of the perceived complexity of the system.

f. All standard software engineering quality metrics.

3.8.2 Recommendations.

3.8.2.1 Subtask A. JLC JPCG-CRM develop and coordinate a security awareness and training program for Project Managers and PDSS operational personnel.

3.8.2.2 Subtask B and C.

a. Strict systems and software engineering standards must be defined and enforced throughout the life cycle (development and post deployment) of the system.

b. Determine the DAA at the beginning and involve the DAA throughout the life cycle of a "secure" system.

c. Risk management must be a continuous process from requirements definition throughout the life cycle.

d. Provide full IV&V documentation to the PDSS Center. This documentation is vital to the post deployment support process.

e. The National Telecommunications and Information System and Security Committee (NTISSC), under the auspices of NSDD-145, must establish a single source for DOD computer security policy. The variety of existing DOD computer security policies and guidelines must be integrated into a single cohesive set, eliminating the confusion caused by conflicting direction.

f. A mechanism for assessing the security impact of a change made to a system must be defined. A necessary part is the placement of the DAA on the configuration control board for the system.

g. The Orange Book requirements must be interpreted and guidance provided to address networks, data bases, and process control security applications as well as information systems security applications other than operating systems.

h. The JLC should establish a committee to develop changes to DOD-STD-2167 that incorporate security requirements as an integral part of a systems development life cycle. The standard must include specific Service requirements as well as National Computer Security Center requirements; it must provide DIDs to detail the required deliverables; and it should be augmented by a guidebook for application of the security standards. (A starting point for such a guideline is the "Computer Security Acquisition Management Guidebook," developed by the Space and Naval Warfare Systems Command.) The basis for this standard should proceed from an appropriate modification to DOD Directives 5000.1 and 5000.2.

i. The Services must have an organic capability to evaluate systems against trusted computing criteria and certify them for the accreditation process. (The certification process provided by the NCSC takes too long.)

j. The JLC should take steps to expedite the development and release of network certification criteria and data base evaluation criteria.

k. The JLC should expedite the completion and release of standard language regarding security requirements for inclusion in contracts and Statements of Work.

l. Establish specific guidelines that address the security requirements for the transition of a system to a PDSS Center.

3.8.2.3 Subtask D.

a. The Government must provide support for verification tools that are to be used for trusted systems development.

b. Establish a Security Efforts Coordination Agent under the JLC JPCG-CRM to make maximum use of individual Service security efforts.

3.8.2.4 Subtask E. Recommendations derived as a result of this subtask were forwarded to Panel V, PDSS Management Indicators and Quality Metrics.

3.8.3 Impacts.

a. Continued use of systems which cannot be trusted to maintain separation of data of different classification or sensitivity, or protect processes from the action of critical or untrusted processes, or which cannot protect critical processes or sensitive data from unauthorized access or tampering by human operators will prevent us from realizing the full potential of available computing capability.

b. We will continue to have to use separate computer systems to process information of different sensitivity levels with the attendant costs of separate, duplicate hardware and the restriction that human interfaces between systems of unequal sensitivity levels have on data sharing among systems.

c. We will continue to have an unacceptable level of confidence when computer systems which control weapons or safety critical devices cannot be disabled or caused to operate other than intended because of the vulnerability of critical processes to other processors, bad data, or out of sequence commands.

3.8.4 Benefits. The R&D recommendations are focused on providing improved tools for satisfying requirements and providing greater assurance that we can trust our security implementations. A longer range recommendation was focused on providing systems architecture to better support security for real-time process control functions. All of the recommendations are aimed primarily at the system acquisition process because it has been shown that attempting to retrofit security is both costly and largely ineffective.

a. Embedding computer security requirements into DOD-STD-2167 will establish computer security as a development discipline across the DOD.

b. Guidance on defining application specific computer security requirements and carrying out computer security functions during the life cycle will support the requirements in DOD-STD-2167.

c. Service organic capabilities for evaluating products against the Trusted Computer Base criteria and certification of the applications systems satisfy selected security requirements will speed up the evaluation and certification process.

The greatest benefits to PDSS activities is for those activities to begin with an accredited system and the tools used to certify that system's compliance with security requirements. These tools are necessary to maintain the system's compliance with its security requirements.

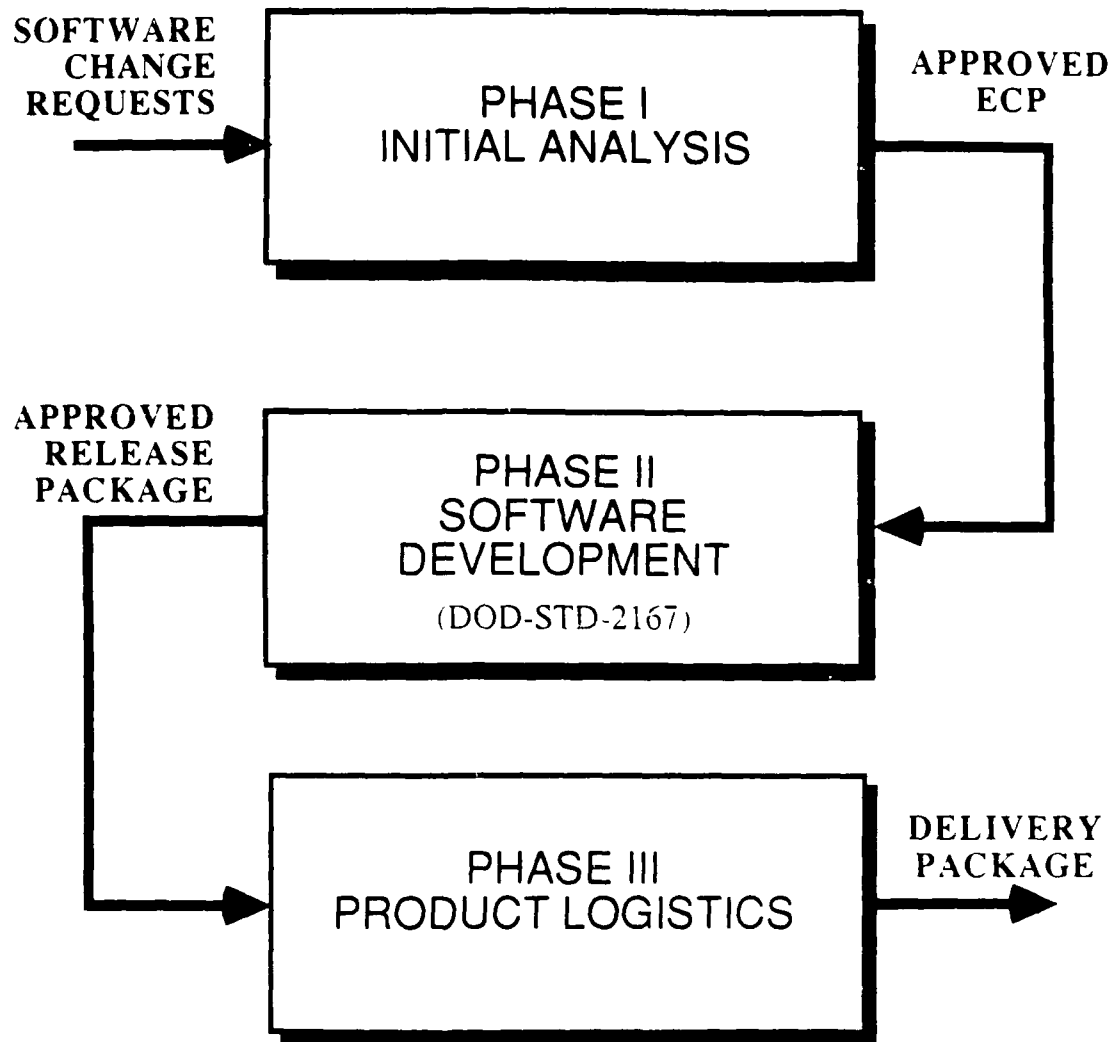


FIGURE 1. PDSS Process

(Intentionally Blank)

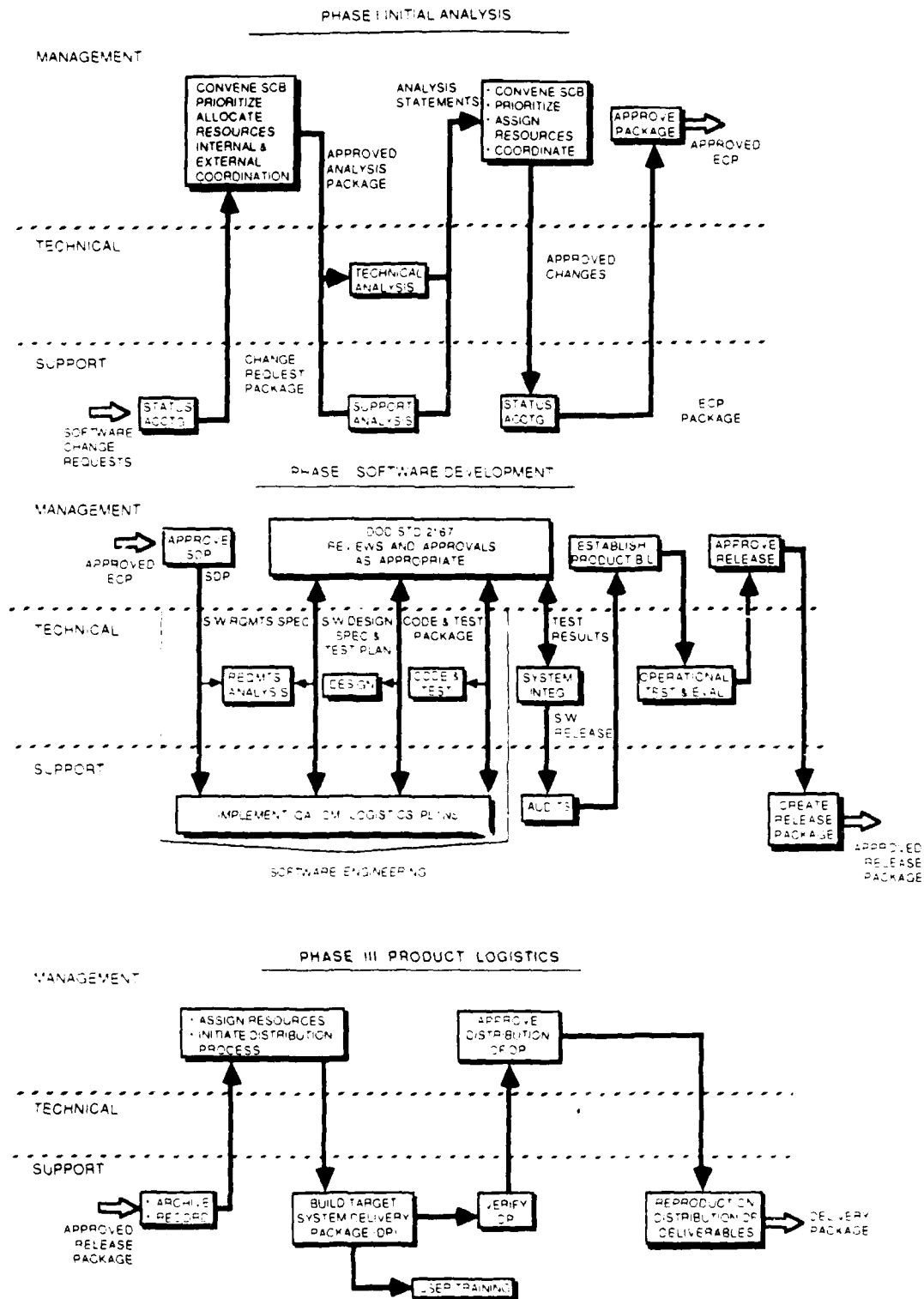


FIGURE 2. PDSS Detailed Model

(Intentionally Blank)

	ACQUISITION PHASE MILESTONES				
	O	I	II	III	SSD
SUPPORT ENVIRONMENT		I	S	-D-	O (SSD - 1 YR)
FACILITIES			I		O (SDD - 1 YR)
PERSONNEL					
TRAINING					
PLANS					
PROCEDURES					
FIELDED SOFTWARE					
MISSION EQUIPMENT					

LEGEND: I = IDENTIFIED, S = SPECIFIED, D = DEVELOPED, O = OPERATIONAL

FIGURE 3. PDSS Planning Activities

(Intentionally Blank)